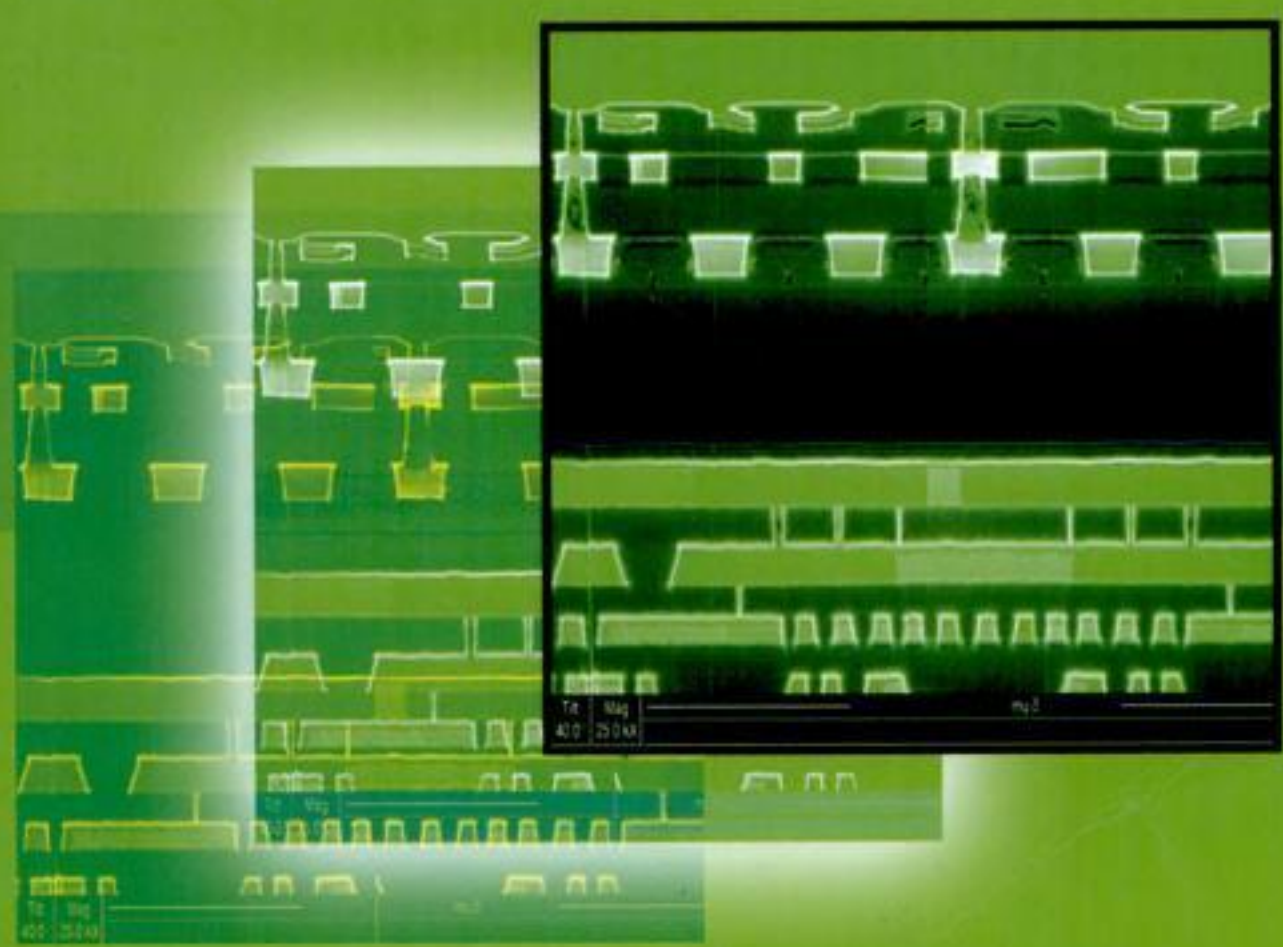
Jan M. Rabaey
Anantha Chandrakasan
Bora Nikolić

CIRCUITI INTEGRATI DIGITALI

L'ottica del progettista

Seconda edizione



Edizione italiana a cura di
Andrea Cester e Andrea Gerosa
Università di Padova

PEARSON
Prentice Hall

Materiale protetto da copyright

Copyright © 2005 Pearson Education Italia S.r.l.
Via Fara, 28 - 20124 Milano
Tel. 02/6739761
Fax 02/673976503
E-mail: hpeitalia@pearson.com
Web: http://hpe.pearsoned.it

Authorized translation from the English language edition, entitled: ***Digital Integrated Circuits, 2nd edition, by Rabaey, Jan M.; Chandrakasan, Anantha; Nikolic, Borivoje*** *published by Pearson Education, Inc, publishing as Prentice Hall, Copyright © 2003*

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc,

Italian language edition published by Pearson Education Italia Srl, Copyright © 2005

Le informazioni contenute in questo libro sono state verificate e documentate con la massima cura possibile. Nessuna responsabilità derivante dal loro utilizzo potrà venire imputata agli Autori, a Pearson Education Italia o a ogni persona e società coinvolta nella creazione, produzione e distribuzione di questo libro.

I diritti di riproduzione e di memorizzazione elettronica totale e parziale con qualsiasi mezzo, compresi i microfilm e le copie fotostatiche, sono riservati per tutti i paesi.

LA FOTOCOPIATURA DEI LIBRI È UN REATO.

L'editore potrà concedere a pagamento l'autorizzazione a riprodurre una porzione non superiore a un decimo del presente volume.
Le richieste di riproduzione vanno inoltrate ad AIDRO (Associazione Italiana per i Diritti di Riproduzione delle Opere dell'Ingegno), Via delle Erbe, 2 - 20121 Milano - Tel. e Fax 02/80.95.06.

Traduzione, revisione scientifica e composizione: Andrea Cester, Andrea Gerosa
Copy-editing: Gioacchino Difazio

Grafica di copertina: Mirella Marinelli
Stampa: Legoprint – Lavis (TN)

Tutti i marchi citati nel testo sono di proprietà dei loro detentori.

ISBN-13: 978-88-7192-231-7
ISBN-10: 88-7192-231-X

Printed in Italy
Settembre 2005

Materiale protetto da copyright

Sommario

Prefazione all'edizione italiana vii

Prefazione alla seconda edizione ix

Parte 1 **Dispositivi e Tecnologie** 2

Capitolo 1 **Introduzione** 3

1.1 Panoramica storica 4
1.2 Problematiche progettuali nei circuiti integrati digitali 7
1.3 Figure di merito di un sistema digitale 17
1.3.1 Costo di un circuito integrato 17
1.3.2 Funzionalità e robustezza 20
1.3.3 Prestazioni 29
1.3.4 Consumo di potenza ed energia 32
1.4 Approfondimenti 34
Bibliografia 34
Riferimenti scientifici 36

Capitolo 2 **Il processo di fabbricazione CMOS** 37

2.1 Introduzione 38
2.2 Fabbricazione dei circuiti integrati CMOS 38
2.2.1 Il wafer di silicio 39
2.2.2 Fotolitografia 40
2.2.3 Principali passi di processo 43
2.2.4 Descrizione esemplificata di un processo CMOS 45
2.3 Regole di layout: il contratto tra il progettista e il costruttore 49
2.4 Il packaging dei circuiti integrati 53
2.4.1 Materiali dei package 56
2.4.2 Livelli di interconnessioni 56
2.4.3 Considerazioni termiche nel packaging 63
2.5 Le prospettive: Evoluzione dei processi tecnologici 65
2.5.1 Nel breve periodo 65
2.5.2 Nel lungo periodo 67
2.6 Riepilogo 68
2.7 Approfondimenti 69
Riferimenti scientifici 69

Materiale protetto da copyright

Metodologie di progetto - Inserto A **Layout dei circuiti digitali** **71**
A.1 Approfondimenti 75
Riferimenti bibliografici 76
Capitolo 3 **I dispositivi** **77**
3.1 Introduzione 78
3.2 Il diodo 79
3.2.1 Un primo sguardo al diodo: la regione di svuotamento 79
3.2.2 Il comportamento statico 82
3.2.3 Il comportamento dinamico o transitorio 85
3.2.4 Il diodo reale: effetti del secondo ordine 88
3.2.5 Il modello SPICE del diodo 90
3.3 Il transistor MOS(FET) 91
3.3.1 Un primo sguardo al dispositivo 92
3.3.2 Caratteristiche statiche del transistor MOS 94
3.3.3 Il comportamento dinamico del transistor MOS 112
3.3.4 Il transistor MOS reale: alcuni effetti secondari 120
3.3.5 Modelli SPICE per il transistor MOS 123
3.4 Una parola sulle variazioni del processo 127
3.5 Prospettive: evoluzione delle tecnologie 129
3.6 Riepilogo 136
3.7 Approfondimenti 137
Riferimenti scientifici 137
Metodologie di progetto - Inserto B **Simulazione circuitale** **139**
Riferimenti scientifici 142
Capitolo 4 **Le connessioni** **145**
4.1 Introduzione 146
4.2 Un primo sguardo 146
4.3 Parametri delle interconnessioni: capacità, resistenza e induttanza 148
4.3.1 Capacità 148
4.3.2 Resistenza 155
4.3.3 Induttanza 159
4.4 Modelli elettrici delle connessioni 161
4.4.1 La connessione ideale 162
4.4.2 Il modello a parametri concentrati 162
4.4.3 Il modello RC a parametri concentrati 163
4.4.4 Il modello rc a parametri distribuiti 167
4.4.5 La linea di trasmissione 171
4.5 I modelli SPICE delle connessioni 182
4.5.1 Le linee a parametri rc distribuiti in SPICE 182
4.5.2 Le linee di trasmissione in SPICE 182

Materiale protetto da copyright

4.5.3 Le prospettive: uno sguardo al futuro 183
4.6 Riepilogo 186
4.7 Approfondimenti 186
Riferimenti scientifici 187

Parte 2 **Il Circuito** **189**

Capitolo 5 **L'invertitore CMOS** **191**
5.1 Introduzione 192
5.2 L'invertitore CMOS statico: Caratteristiche fondamentali 192
5.3 Robustezza dell'invertitore CMOS: il comportamento statico 197
5.3.1 Soglia di commutazione 197
5.3.2 Margini di rumore 200
5.3.3 Robustezza e tecnologia 204
5.4 Prestazioni dell'invertitore CMOS: il comportamento dinamico 207
5.4.1 Il calcolo delle capacità 207
5.4.2 Il tempo di propagazione: analisi del primo ordine 213
5.4.3 Ottimizzazione del tempo di propagazione 217
5.5 Potenza, energia e prodotto energia-ritardo 228
5.5.1 Il consumo dinamico di potenza 229
5.5.2 Il consumo statico di potenza 239
5.5.3 Il consumo totale 241
5.5.4 Analisi del consumo di potenza con SPICE 244
5.6 Prospettive: Evoluzione della tecnologia e figure di merito dell'invertitore 246
5.7 Riepilogo 248
5.8 Approfondimenti 250
Riferimenti scientifici 250

Capitolo 6 **Porte logiche combinatorie CMOS** **251**
6.1 Introduzione 252
6.2 Logica CMOS statica 252
6.2.1 Logica CMOS statica complementare 253
6.2.2 Logica a rapporto 282
6.2.3 Logica pass-transistor 289
6.3 Logica CMOS dinamica 305
6.3.1 Principi base 306
6.3.2 Velocità e consumo di potenza delle porte logiche dinamiche 308
6.3.3 Robustezza e integrità del segnale nella logica dinamica 312
6.3.4 Porte dinamiche in cascata 319

Materiale protetto da copyright

6.4 Prospettive 327
6.4.1 Come scegliere il tipo di famiglia logica 327
6.4.2 La diminuzione della tensione di alimentazione 327
6.5 Riepilogo 330
6.6 Approfondimenti 331
Riferimenti scientifici 332

Metodologie di progetto - Inserto C **La simulazione di circuiti logici complessi 333**
C.1 Rappresentazione dei dati digitali con quantità continue 334
C.2 Rappresentazione dei dati digitali con quantità discrete 334
C.3 Usare modelli ad alto livello 339
Riferimenti scientifici 342

Metodologie di progetto - Inserto D **Layout di porte logiche complesse 343**

Capitolo 7 **Circuiti logici sequenziali 349**
7.1 Introduzione 350
7.1.1 Caratteristiche temporali dei circuiti sequenziali 350
7.1.2 Classificazione degli elementi di memoria 352
7.2 Latch e registri statici 354
7.2.1 Il principio di bistabilità 354
7.2.2 Il latch a multiplexer 357
7.2.3 Registri master–slave ed edge-triggered 358
7.2.4 Latch statici a bassa tensione 364
7.2.5 Flip-flop SR statici: memorizzare i dati per "forza bruta" 366
7.3 Latch e registri dinamici 368
7.3.1 Registri dinamici a gate di trasmissione 369
7.3.2 C^2MOS—una soluzione insensibile al clock-skew 372
7.3.3 I registri a fase di clock singola 375
7.4 Altri tipi di registri* 380
7.4.1 Registri con clock a impulsi 380
7.4.2 Registri a Sense-amplifier 383
7.5 Pipeline: l'ottimizzazione dei circuiti sequenziali 385
7.5.1 Pipeline con latch o con registri 386
7.5.2 NORA–CMOS: un esempio di architettura pipeline 387
7.6 Altri circuiti sequenziali 389
7.6.1 Il trigger di Schmitt 389
7.6.2 I circuiti monostabili 393
7.6.3 I circuiti astabili 394
7.7 Le prospettive: scelta della struttura di temporizzazione 398
7.8 Riepilogo 398
7.9 Approfondimenti 399

Materiale protetto da copyright

Riferimenti scientifici 399

Parte 3 **Il Sistema** **401**

Capitolo 8 **Tecniche di implementazione dei sistemi digitali** **403**
8.1 Introduzione 404
8.2 Dall'approccio dedicato a quello semi-automatico a celle standard 409
8.3 La progettazione dedicata 410
8.4 Metodologie progettuali basate su celle 410
8.4.1 Celle standard 411
8.4.2 Celle compilate 417
8.4.3 Macrocelle, megacelle e proprietà intellettuale 419
8.4.4 Flusso di progetto semi-automatico 423
8.5 Implementazione a matrici logiche 426
8.5.1 Matrici logiche senza interconnessioni 427
8.5.2 Matrici a interconnessioni programmabili 432
8.6 Le prospettive: le piattaforme di implementazione del futuro 449
8.7 Approfondimenti 452
Riferimenti scientifici 452

Metodologie di progetto - Inserto E **Caratterizzazione delle celle combinatorie e sequenziali** **455**
Riferimenti scientifici 462

Metodologie di progetto - Inserto F **Sintesi del circuito** **463**
Riferimenti scientifici 471

Capitolo 9 **Interconnessioni e parassiti** **473**
9.1 Capacità parassite 474
9.1.1 Capacità e robustezza: cross talk 474
9.1.2 Capacità e prestazioni dei circuiti CMOS 477
9.2 Resistenze parassite 488
9.2.1 Resistenza e robustezza: cadute di tensione 488
9.2.2 Elettromigrazione 491
9.2.3 Resistenza e prestazioni: ritardo RC 493
9.3 Induttanze parassite* 498
9.3.1 Induttanze e robustezza: cadute di tensione 499
9.3.2 Induttanze e prestazioni: linee di trasmissione 505
9.4 Tecniche di interconnessione avanzate 509
9.4.1 Circuiti a escursione logica ridotta 510
9.4.2 Tecniche di trasmissione nel dominio della corrente 517

Materiale protetto da copyright

9.5 Prospettive: reti su singolo chip 518
9.6 Riepilogo 519
9.7 Approfondimenti 520
Riferimenti scientifici 520

Capitolo 10 **Temporizzazione e sincronismo nei circuiti digitali 523**
10.1 Introduzione 524
10.2 Classificazione della temporizzazione di un sistema digitale 524
10.2.1 Interconnessione sincrona 525
10.2.2 Interconnessione mesocrona 525
10.2.3 Interconnessione plesiocrona 526
10.2.4 Interconnessione asincrona 527
10.3 Sistemi sincroni: approfondimento 527
10.3.1 Principi della temporizzazione sincrona 527
10.3.2 Origine di skew e jitter 535
10.3.3 Tecniche di distribuzione del clock 542
10.3.4 Sincronizzazione dei circuiti basata su latch* 550
10.4 Circuiti autotemporizzati* 554
10.4.1 Logica autotemporizzata: una tecnica asincrona 554
10.4.2 Generazione dei segnali di completamento 557
10.4.3 Segnali di controllo nei circuiti autotemporizzati 561
10.4.4 Esempi di logica autotemporizzata 567
10.5 Sincronizzatori e arbitri* 570
10.5.1 Sincronizzatori: concetti e realizzazione 570
10.5.2 Arbitri 574
10.6 Generazione del clock e sincronizzazione con PLL* 575
10.6.1 Concetti fondamentali 576
10.6.2 Blocchi fondamentali di un PLL 578
10.7 Sviluppi e prospettive futuri 583
10.7.1 Clock distribuiti e DLL 583
10.7.2 Distribuzione ottica del clock 585
10.7.3 Confronto tra sistemi sincroni e asincroni 586
10.8 Riepilogo 587
10.9 Approfondimenti 587
Riferimenti scientifici 588

Metodologie di progetto - Inserto G **Verifica del progetto 591**
Riferimenti scientifici 596

Capitolo 11 **Blocchi logici e aritmetici 597**
11.1 Introduzione 598
11.2 Architettura dell'unità logico-aritmetica

Materiale protetto da copyright

dei microprocessori 598
11.3 I sommatori 599
11.3.1 Definizioni generali 600
11.3.2 Ottimizzare del circuito full-adder 603
11.3.3 Ottimizzare l'architettura di un sommatore 610
11.4 I moltiplicatori 627
11.4.1 L'operazione di moltiplicazione binaria 627
11.4.2 Calcolo dei prodotti parziali 628
11.4.3 Accumulazione dei prodotti parziali 631
11.4.4 L'addizione finale 636
11.4.5 Riepilogo 636
11.5 I traslatori 637
11.5.1 Traslatore barrel 637
11.5.2 Traslatore logaritmico 639
11.6 Altri operatori aritmetici 640
11.7 Compromesso tra velocità e consumo nelle unità di elaborazione * 643
11.7.1 Tecniche di riduzione del consumo in fase di progetto 644
11.7.2 Gestione del consumo durante il funzionamento 656
11.7.3 Riduzione del consumo statico in fase di inattività (standby) 662
11.8 Prospettive: scelte e compromessi nella progettazione 664
11.9 Riepilogo 665
11.10 Approfondimenti 666
Riferimenti scientifici 667

***Capitolo 12* Memorie a semiconduttore 669**
12.1 Introduzione 670
12.1.1 Classificazione delle memorie 671
12.1.2 Architettura e blocchi funzionali delle memorie 674
12.2 Il nucleo della memoria: la matrice di celle 681
12.2.1 Memorie a sola lettura 681
12.2.2 Memorie non volatili 697
12.2.3 Memorie a lettura-scrittura (RAM) 708
12.2.4 Memorie Associative (CAM) 723
12.3 Circuiti periferici delle memorie* 725
12.3.1 Decodificatore di indirizzo 725
12.3.2 Sense Amplifier 733
12.3.3 Generazione delle tensioni di riferimento 740
12.3.4 Buffer 744
12.3.5 Temporizzazione e controllo 745
12.4 Robustezza, affidabilità e resa* 748
12.4.1 Rapporto segnale-rumore 748
12.4.2 Resa 755

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PARTE

1

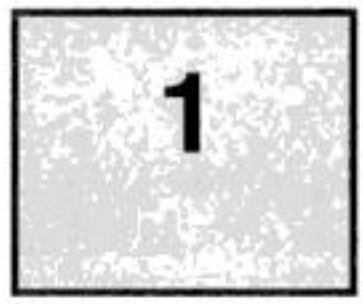
Dispositivi e Tecnologie

"La densità di componenti è raddoppiata ogni anno. Sicuramente, a breve termine ci si può attendere che questo ritmo continui, o addirittura acceleri. Nel lungo periodo, la velocità è meno prevedibile, sebbene non vi sia alcuna ragione per credere che questo ritmo non debba rimanere circa costante per almeno dieci anni. Ciò significa che nel 1975, il numero di componenti per circuito integrato sarà 65 000."

Gordon Moore,
Cramming more Components onto Integrated Circuits, (1965).

Materiale protetto da copyright

CAPITOLO



Introduzione

Evoluzione della progettazione dei circuiti digitali

Problematiche progettuali nei circuiti digitali

Figure di merito di un sistema digitale

Riferimenti bibliografici

1.1 Panoramica storica
1.2 Problematiche progettuali nei circuiti integrati digitali
1.3 Figure di merito di un sistema digitale
  1.3.1 Costo di un circuito integrato
  1.3.2 Funzionalità e robustezza
  1.3.3 Prestazioni
  1.3.4 Consumo di potenza ed energia
1.4 Riepilogo
1.5 Approfondimenti

Materiale protetto da copyright

1.1 Panoramica storica

Il concetto di elaborazione dei dati digitali ha avuto un forte impatto nella nostra società, tanto che ognuno di noi è ormai abituato all'idea stessa di computer. In continua evoluzione dai *mainframe* ai minicomputer, i personal computer e i computer portatili proliferano e si diffondono nella vita di tutti i giorni. Ancor più significativa è la spinta verso l'impiego di soluzioni digitali in tutti i campi dell'elettronica. La strumentazione di misura fu una delle prime applicazioni, non collegate direttamente ai computer, dove furono riconosciuti i potenziali benefici dell'elaborazione digitale dei dati rispetto al convenzionale trattamento analogico dei segnali. Molte altre applicazioni in aree diverse, quale i controlli, presto seguirono la stessa strada. Solo recentemente siamo stati testimoni della conversione delle telecomunicazioni e dell'elettronica di consumo al mondo digitale. Sempre più i dati telefonici sono trasmessi ed elaborati in modo digitale su reti con e senza fili. I compact disc hanno rivoluzionato le applicazioni audio, e il video digitale ha ormai intrapreso la stessa strada.

L'idea realizzare di sistemi computazionali utilizzando un formato di dati codificato non è affatto un'idea dei nostri tempi. Nei primi anni del diciannovesimo secolo, Charles Babbage previde la realizzazione di sistemi computazionali meccanici su larga scala, che chiamò *Macchine Differenziali* (*Difference Engine*) [Swade93]. Sebbene tali macchine utilizzassero il sistema di numerazione decimale anziché la rappresentazione binaria attualmente diffusa nell'elettronica moderna, i concetti di base erano molto simili. La *Macchina Analitica (Analytical Engine)*, sviluppata nel 1834, fu concepita come sistema computazionale d'uso generale con caratteristiche del tutto simili ai moderni computer. Oltre ad eseguire il consueto repertorio di operazioni (addizione, sottrazione, moltiplicazione e divisione) in sequenza arbitraria, la macchina lavorava in una sequenza costituita di due cicli, chiamati "store" (memorizzazione) e "mill" (esecuzione), non dissimili da quelli usati dagli elaboratori di oggi. La macchina impiegava anche l'architettura *pipeline* per accelerare l'esecuzione dell'operazione di somma. Sfortunatamente, la complessità e i costi di progettazione resero tali concetti irrealizzabili. Il progetto della prima versione del calcolatore differenziale (parte del quale è mostrato in Figura 1-1), per esempio, prevedeva 25 000 parti meccaniche e un costo complessivo di 17 470 sterline del 1834.

La soluzione elettrica si mostrò essere più efficiente in termini di costi. I primi sistemi elettronici erano basati su interruttori a controllo magnetico (o relè). Essi erano usati principalmente per la realizzazione di reti logiche molto semplici, e attualmente sono impiegati nei sistemi di sicurezza dei treni. L'era dei veri e propri sistemi di calcolo elettronici cominciò in pieno solo con l'introduzione dei tubi a vuoto o valvole termoioniche (*vacuum tube*). Originariamente usati quasi esclusivamente per il trattamento dei segnali analogici, fu presto riconosciuta la loro utilità anche per l'elaborazione digitale. Di lì a poco, fu realizzato il primo computer completo. L'epoca degli elaboratori basati su valvole culminò con la progettazione e la realizzazione di macchine quali l'ENIAC (ideato per calcolare i tabulati relativi alle piattaforme di artiglieria) e l'UNIVAC I (il primo computer commerciale di successo). Per dare un'idea della densità di integrazione, l'ENIAC era lungo più di 24 metri, alto 2.6 metri, largo parecchi metri ed

Materiale protetto da copyright

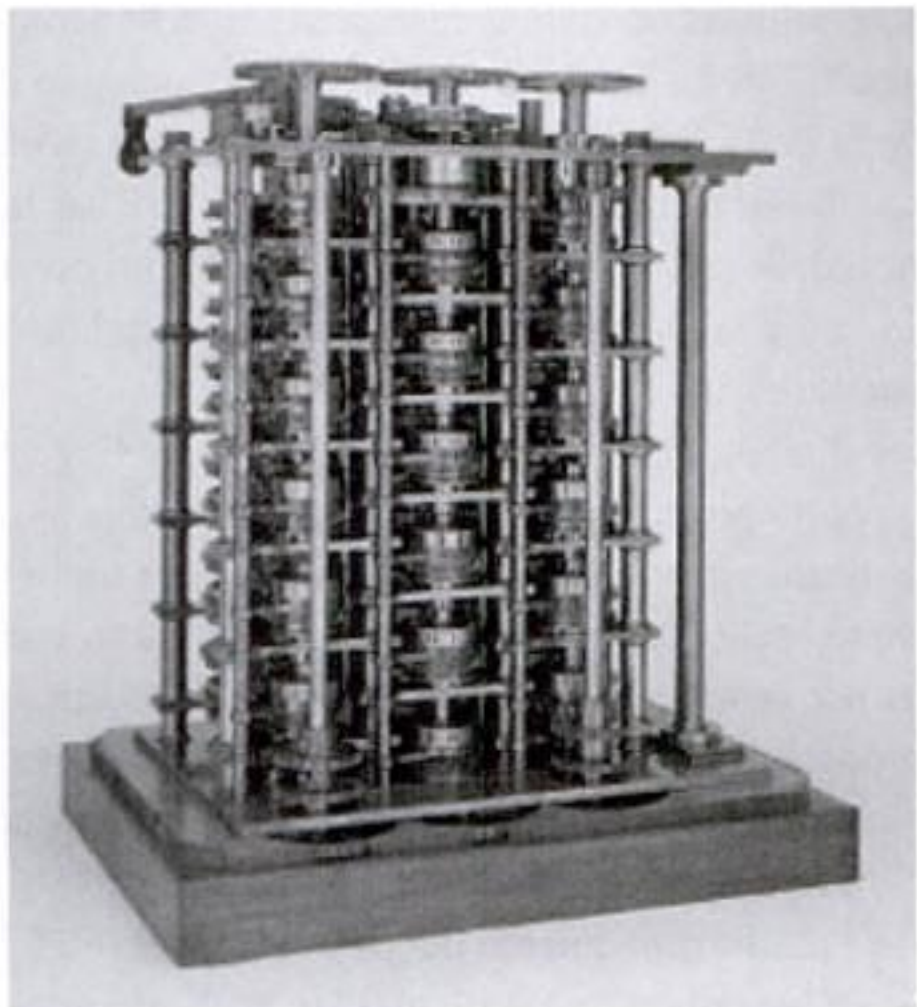


Figura 1-1 Dettaglio della prima macchina differenziale di Babbage, il primo calcolatore automatico conosciuto (tratto da [Swade93], gentile concessione del Science Museum di Londra).

era costruito con 18 000 valvole. Divenne però subito chiaro che questa tecnologia aveva raggiunto il suo limite. A causa di problemi di affidabilità e dell'eccessivo consumo di potenza, l'implementazione di macchine ancor più grandi e sofisticate fu una cosa improponibile sia sul piano economico sia su quello pratico.

Lo scenario mutò completamente con l'invenzione del *transistor* ai Bell Telephone Laboratories nel 1947 [Bardeen48], seguito dall'introduzione del transistor bipolare (*bipolar junction transistor*, BJT) da parte di Schockley nel 1949 [Schockley49].[1] Fu necessario, però, attendere fino al 1956 perché il primo circuito logico in tecnologia bipolare, costituito di componenti discreti, fosse introdotto da Harris [Harris56]. Nel 1958, alla Texas Instruments, Jack Kilby, concepì l'idea di circuito integrato *(IC, integrated circuit)*, nel quale tutti i componenti, passivi e attivi, erano integrati nello stesso substrato di semiconduttore. Per questa idea incredibilmente innovativa gli fu conferito il premio Nobel. Ciò condusse all'introduzione della prima famiglia di porte logiche commerciali su circuito integrato, denominata *Fairchild Micrologic family* [Norman60]. I primi circuiti integrati che ebbero un vero e proprio successo furono quelli appartenenti alla famiglia logica *TTL (Transistor-Transistor Logic)*, introdotta nel 1962 [Beeson62]; successivamente furono sviluppate altre famiglie logiche con prestazioni ancora migliori. Un esempio sono i circuiti a commutazione di corrente che condussero alla fabbricazione delle prime porte logiche con tempi di commutazione

[1] Un'interessante panoramica sull'evoluzione dei circuiti integrati digitali si può trovare in [Murphy93] (la maggior parte dei dati presentati in questa introduzione sono stati tratti proprio da quel lavoro). In tale lavoro sono citate alcune tra le pubblicazioni storicamente più significative nel campo dei circuiti integrati.

Materiale protetto da copyright

inferiori al nanosecondo e culminarono con la realizzazione della famiglia logica *ECL (Emitter-Coupled Logic)* [Masaki74]. La famiglia TTL aveva il vantaggio di offrire una maggior densità di integrazione e fu la base di partenza della prima grande rivoluzione nella storia dei circuiti integrati. La possibilita di fabbricare i componenti TTL su larga scala fu il fattore determinante nel successo delle prime industrie di semiconduttori come Fairchild, National e Texas Instruments. Tale famiglia logica ebbe un tale successo da costituire la maggiore fetta del mercato dei semiconduttori fino agli anni '80.

Alla fine, la logica digitale bipolare perse la battaglia per l'egemonia nel mondo della progettazione digitale, proprio perché aveva ereditato dalle valvole lo stesso principio di funzionamento: l'elevato consumo di potenza per porta logica pose un limite al numero di porte logiche che possono essere realisticamente integrate in un singolo chip o sistema. Anche se furono fatti molti sforzi per sviluppare famiglie logiche, basate sempre su tecnologia bipolare, che permettessero una elevata integrazione e un basso consumo (ad esempio la I^2L — *Integrated Injection Logic* [Hart72]), alla fine trionfarono i circuiti integrati basati sulla tecnologia MOS.

Il principio alla base del funzionamento del transistor MOSFET (originariamente chiamato IGFET) fu proposto e brevettato da J. Lilienfeld (Canada) già nel 1925 e, indipendentemente, da O. Heil in Inghilterra nel 1935. Le conoscenze insufficienti sui materiali e problemi legati alla stabilità del gate ritardarono di molto l'effettivo impiego di tale dispositivo. Non appena i problemi furono risolti, la diffusione dei circuiti integrati basati sulla tecnologia MOS decollò a pieno regime; ma ciò avvenne soltanto a partire dai primi anni '70. In particolare, vale la pena ricordare che le prime porte logiche MOS ad essere introdotte erano di tipo CMOS [Wanlass63], e questa tendenza continuò fino alla fine degli anni '60. La complessità dei processi manifatturieri ritardò la piena espansione di questi dispositivi ancora per due decenni. Il primo circuito digitale integrato di tipo MOS fu realizzato in logica PMOS (utilizzando solo PMOS) e fu impiegato in applicazioni quali le calcolatrici. La seconda era della rivoluzione dei circuiti integrati fu inaugurata con l'introduzione del primo microprocessore da parte di Intel nel 1972 (il 4004) [Faggin72] e nel 1974 (l'8080) [Shima74]. Questi processori erano fabbricati in logica NMOS, più veloce rispetto alla logica PMOS. Contemporaneamente, la tecnologia MOS permise la fabbricazione delle prime memorie a semiconduttore ad alta densità d'integrazione. Per esempio, nel 1970 fu introdotta la prima memoria MOS a 4Kbit [Hoff70].

Tutti questi eventi furono l'inizio della stupefacente evoluzione dei circuiti integrati verso sempre maggiori valori di densità di integrazione, velocità e prestazioni; una rivoluzione che tutt'oggi sta marciando a pieni giri. Tuttavia, la corsa verso gli attuali livelli di integrazione non è stata priva di ostacoli. Alla fine degli anni '70, la logica NMOS iniziò a soffrire dello stesso male che rese poco attraente, se non impraticabile, l'elevata integrazione della logica bipolare: il consumo di potenza. Questi problemi, in combinazione con i progressi nelle tecnologie di fabbricazione, fecero definitivamente pendere l'ago della bilancia verso la tecnologia CMOS, e lì si trova ancora oggi. Un fatto abbastanza interessante è che il

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

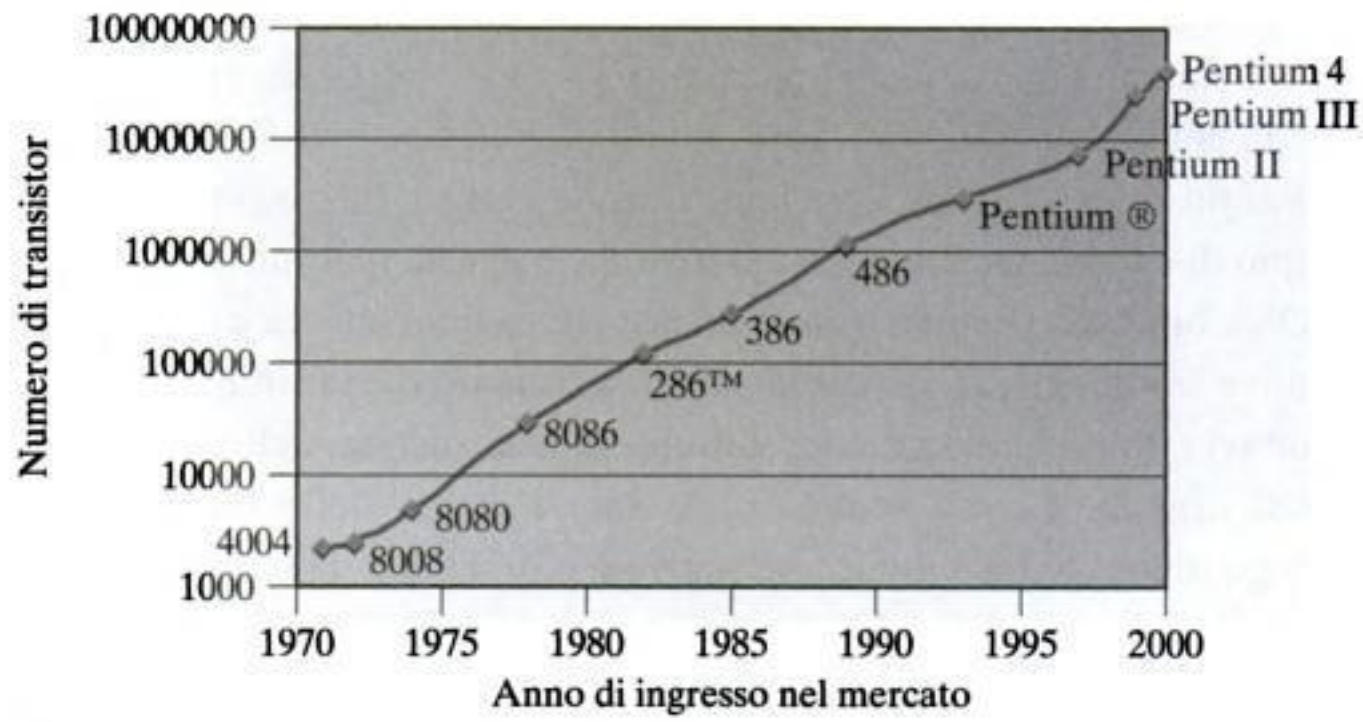


Figura 1-3 Evoluzione storica della densità di integrazione dei microprocessori (da [Intel01]).

Un tipico esempio è offerto dai microprocessori: a partire dalla loro introduzione negli anni '70, essi sono cresciuti in prestazioni e complessità ad un passo costante e prevedibile. La Figura 1-3 mostra l'evoluzione del numero di transistor contenuti all'interno di alcuni ben noti microprocessori, utilizzati qui come riferimento storico. La barriera del milione di transistor per chip fu raggiunta e oltrepassata alla fine degli anni '80, mentre la frequenza di clock è raddoppiata ogni tre anni circa nel corso dell'ultimo decennio raggiungendo valori intorno al GHz. Ciò è illustrato in Figura 1-4, che raffigura l'evoluzione delle prestazioni dei microprocessori fino agli inizi del XXI secolo. È importante notare che fino ad oggi questo continuo incremento di prestazioni non ha dato segni di rallentamento.

Non deve sorprendere il fatto che questa rivoluzione abbia avuto grosse conseguenze sulle metodologie di progetto dei circuiti integrati. I primi progetti erano veramente "disegnati a mano": ciascun transistor era posizionato e ottimizzato individualmente adattandolo a ciò che gli stava intorno. Ciò è adeguatamente illustrato in Figura 1-5a, che mostra una foto-

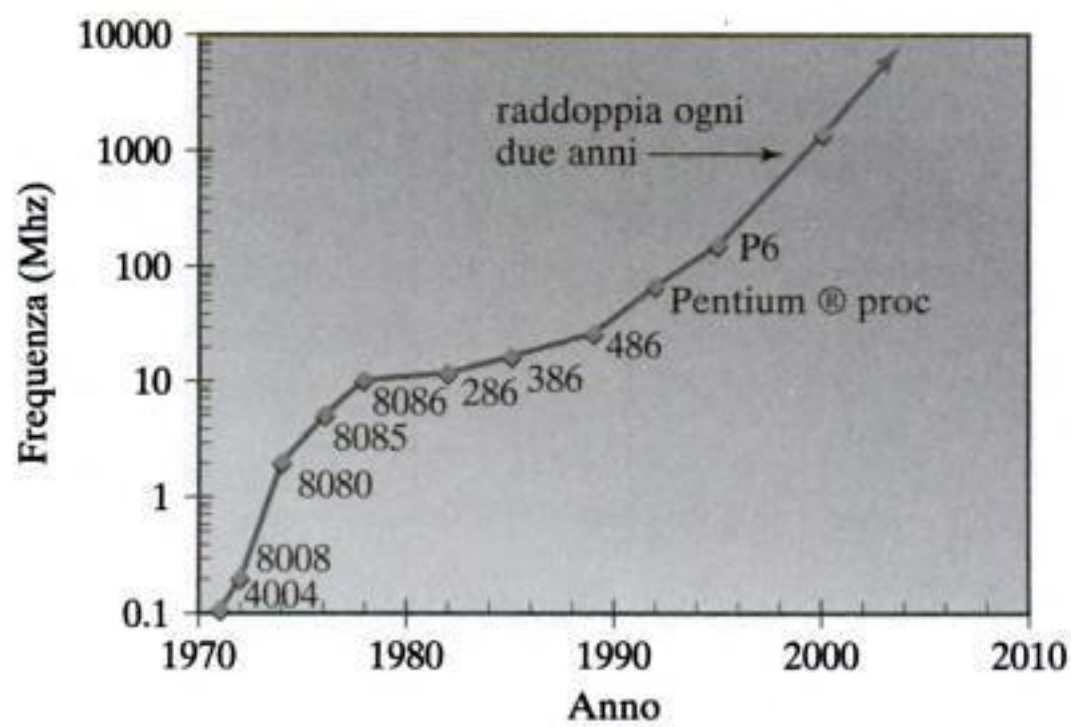
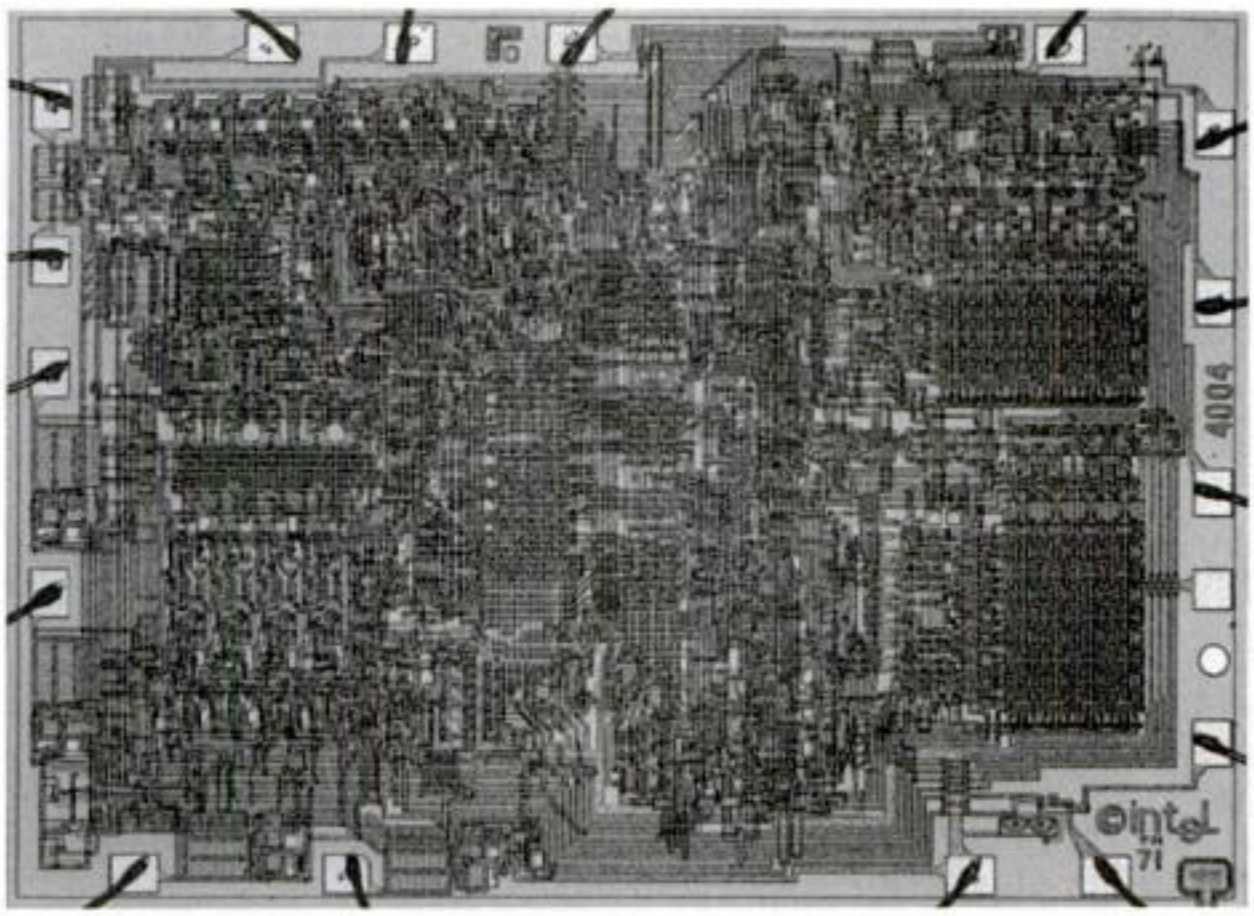
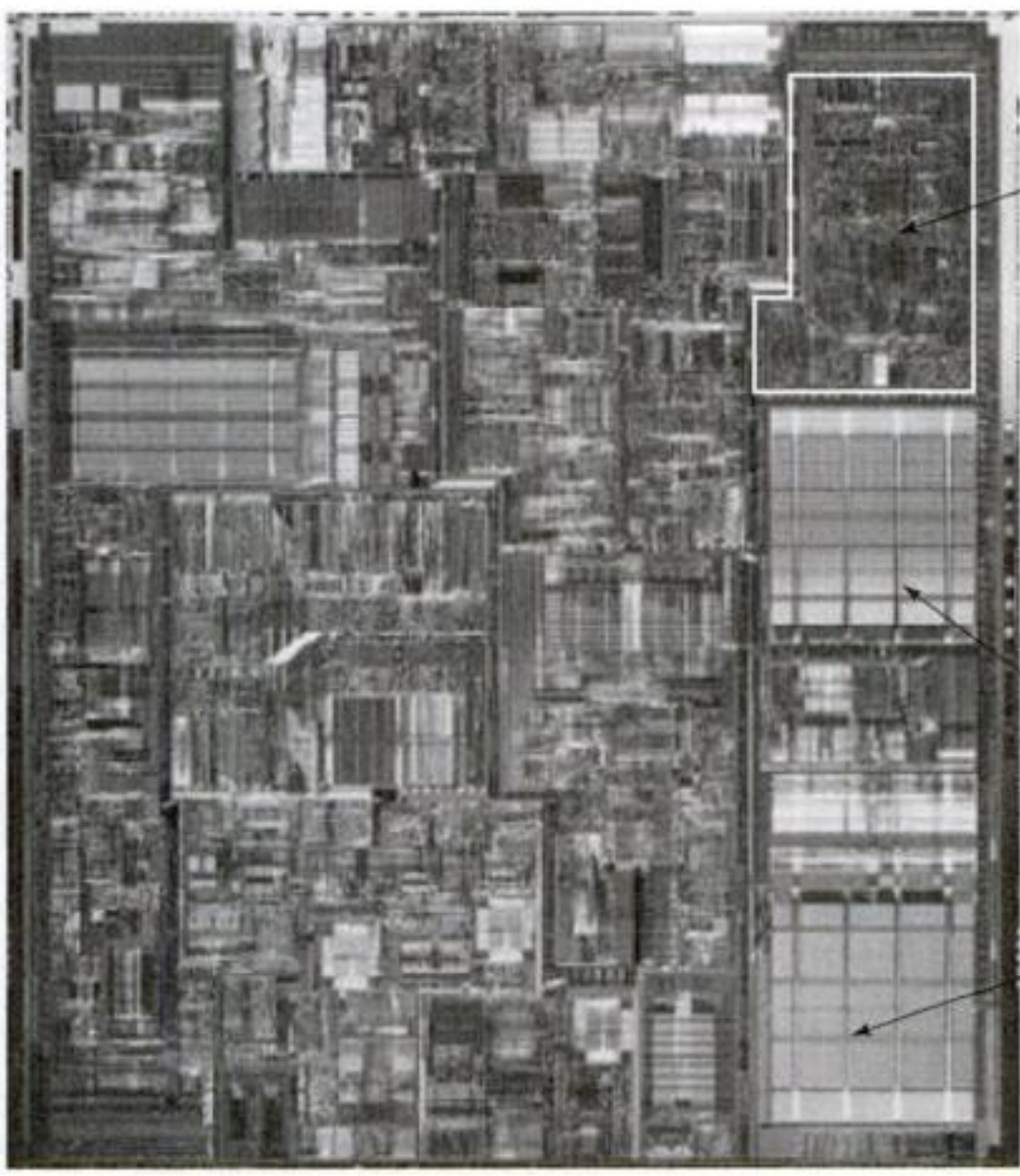


Figura 1-4 Evoluzione delle prestazioni dei microprocessori. (gentile concessione di Intel.)

Materiale protetto da copyright



(a) Il microprocessore Intel 4004



(b) Il processore Intel Pentium ® 4

Figura 1-5 Confronto tra la metodologia di progetto dei processori Intel 4004 (1971) e Pentium® 4 (2000) (gentile concessione di Intel).

Materiale protetto da copyright

grafia del processore Intel 4004. Ovviamente, questo tipo di approccio è impensabile quando è necessario disegnare e collocare un milione e più transistor. Con il rapido sviluppo delle tecnologie progettuali, uno dei fattori chiave per il successo di un circuito è il tempo necessario per la sua immissione sul mercato.

Conseguentemente, i progettisti furono obbligati sempre più ad attenersi a metodologie e strategie progettuali che portarono pian piano verso la progettazione automatizzata. Il risultato di questo nuovo approccio è evidente nel layout di uno degli ultimi microprocessori intel, il Pentium® 4, illustrato in Figura 1-5b. Contrariamente all'approccio personalizzato dei primi progettisti, il circuito è costruito con uno schema gerarchico: un processore è costituito da un insieme di moduli, ciascuno dei quali è suddiviso a sua volta in un certo numero di celle. Le varie celle sono utilizzate e ripetute il più possibile sia per ridurre il lavoro di progettazione sia per aumentare le probabilità che il circuito sia progettato correttamente fin dalla prima versione. La possibilità di usare questa tecnica di progettazione modulare e gerarchica è stato l'ingrediente di base che ha consentito il successo dei circuiti digitali, ma, d'altro canto, spiega perché la progettazione di circuiti analogici VLSI sia una strada difficilmente percorribile.

La domanda immediatamente successiva è: perché questo tipo di approccio è compatibile con il mondo digitale, ma non (o poco) con la progettazione di circuiti analogici? Quando abbiamo a che fare con il problema della complessità, il concetto fondamentale è l'*astrazione*. A ciascun livello di un progetto, i dettagli interni di un modulo complesso si possono astrarre e il modulo si può sostituire con una "scatola nera" o *modello*. Questo modello contiene virtualmente tutte le informazioni per interfacciare il modulo con gli altri blocchi al successivo livello gerarchico di astrazione. Ad esempio, una volta che un progettista ha realizzato un blocco moltiplicatore, le sue caratteristiche e prestazioni possono essere definite molto accuratamente e sintetizzate in un modello. In generale, il comportamento del moltiplicatore è solo marginalmente influenzato dal modo in cui esso sarà utilizzato in un sistema più grande e complesso. Quindi, può essere considerato come una "scatola nera" con caratteristiche note. Nel caso in cui non vi sia alcuna particolare esigenza da parte del progettista di "guardare all'interno della scatola", la complessità del progetto risulta sostanzialmente ridotta. Questo approccio di tipo "*divide et impera*" ha importanti riflessi: anziché avere a che fare con una moltitudine di elementi, il progettista deve considerare solo un manciata di componenti, ciascuno dei quali è descritto da un ridotto numero di parametri.

La stessa cosa accade a un programmatore che utilizzi una libreria di funzioni standard, per esempio i driver di input/output: a chi scrive un programma lungo e complesso non interessa guardare i dettagli di tutte le funzioni delle librerie. Le sole informazioni di interesse sono i risultati prodotti da una chiamata a una determinata funzione. Basti pensare a quanto sarebbe pesante scrivere dei programmi se ci si dovesse preoccupare della lettura delle informazioni bit per bit dall'hard disk e verificarne la correttezza, anziché utilizzare comandi semplici e maneggevoli come "file open" e "get string".

I livelli di astrazione tipicamente utilizzati nella progettazione di circuiti digitali sono, in ordine di astrazione crescente, il dispositivo, il circuito, la porta logica, il modulo o blocco

Materiale protetto da copyright

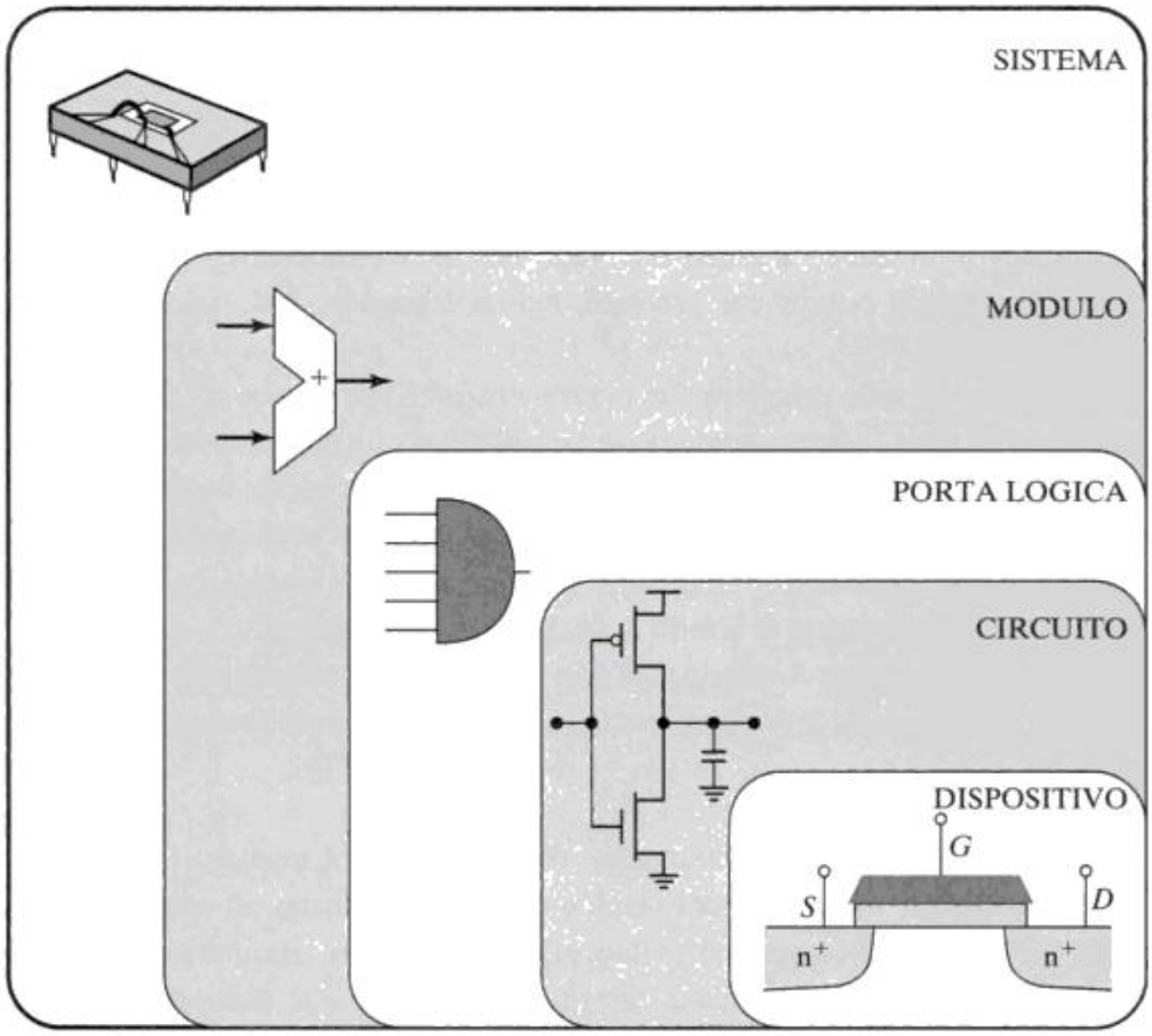


Figura 1-6 Livelli di astrazione nella progettazione di circuiti integrati digitali.

funzionale (per esempio, un sommatore) e il sistema (per esempio, un processore), come illustrato in Figura 1-6. Un dispositivo a semiconduttore è un'entità con un comportamento molto complesso: realisticamente nessun progettista circuitale considererà mai le equazioni fisiche che descrivono il comportamento del dispositivo, mentre sta progettando una porta logica. Egli userà, invece, un modello semplificato che descrive in modo soddisfacente le caratteristiche ingresso/uscita del transistor. Ad esempio, una porta AND è adeguatamente descritta dalla sua espressione booleana ($Z = A \cdot B$), il rettangolo che la contiene, la posizione dei terminali di ingressi e di uscita e il tempo di ritardo tra ingressi e uscita.

Questa filosofia di progetto ha consentito la nascita della progettazione assistita dal calcolatore (*computer-aided design*, CAD) dei circuiti integrati digitali e senza di essa l'attuale complessità circuitale non sarebbe mai stata raggiunta. Gli strumenti di progettazione comprendono la simulazione a diversi livelli di complessità, la verifica del progetto, la generazione del layout e la sintesi del progetto. Una panoramica generale di questi strumenti e metodologie sarà fornita nel Capitolo 8.

Per evitare di riprogettare e riverificare la correttezza delle celle utilizzate più frequentemente, come le porte logiche fondamentali, i blocchi aritmetici e i banchi di memoria, il progettista fa spesso ricorso a *librerie di celle*. Queste librerie contengono non solo il layout,

Materiale protetto da copyright

ma anche una completa caratterizzazione e documentazione del comportamento delle celle stesse. Un esempio di impiego di librerie di celle è fornito dal *layout* del processore Pentium® 4 (Figura 1-5b). Per esempio le *unità aritmetiche intera* e a *virgola mobile (floating point)*: contengono ampie sezioni disegnate usando questo approccio modulare, detto anche "a celle standard" *(standard cell)*. Le porte logiche sono disposte in file di celle di uguale altezza e collegate tra loro utilizzando un'area riservata alle interconnessioni *(routing channel)*. Il layout di questi blocchi può essere generato automaticamente, non appena sia disponibile un'adeguata libreria di celle.

L'analisi fin qui fatta dimostra che la progettazione automatica e l'impiego pratico di blocchi modulari hanno concretamente risolto molti dei problemi di complessità degli attuali circuiti integrati. Ciò conduce alla domande seguenti: se la progettazione automatica risolve tutti i nostri problemi, perché dovremmo preoccuparci di progettare circuiti digitali? La prossima generazione di progettisti digitali sarà ancora interessata ai transistor e agli elementi parassiti o, piuttosto, si fermerà al livello di porta logica e modulo funzionale?

La realtà è molto più complessa e ci sono diverse ragioni per le quali saper guardare all'interno dei circuiti e alla loro complessità sarà ancora per molto tempo un fattore essenziale:

- Innanzitutto, qualcuno deve comunque disegnare e implementare i moduli contenuti nelle librerie. Le tecnologie dei semiconduttori continuano ad avanzare anno dopo anno. Fino a quando qualcuno svilupperà un approccio sistematico e affidabile che renda possibile tradurre una data libreria di celle da una tecnologia all'altra, ciascun cambiamento tecnologico (che avviene approssimativamente ogni due anni) richiede di riprogettare o adattare le librerie.
- Creare un modello adeguato di una cella o di un modulo richiede una profonda conoscenza della sua struttura interna e del suo funzionamento. Per esempio, per identificare i parametri funzionali dominanti di un dato circuito, è necessario conoscere il cammino critico del segnale dall'ingresso all'uscita.
- L'approccio basato su libreria funziona bene quando i vincoli progettuali (velocità, costo o consumo di potenza) non sono stringenti. Questo è il caso di tutti quei circuiti progettati per una specifica applicazione *(application-specific design)*, dove l'obiettivo principale è quello di ottenere una maggiore integrazione di sistema, mentre le prestazioni richieste sono facilmente raggiunte grazie alle potenzialità della tecnologia. Sfortunatamente, in molti altri casi, come ad esempio i microprocessori, il successo del prodotto dipende dalle elevate prestazioni; quindi i progettisti tendono a spingere la tecnologia ai suoi limiti. In questo caso l'approccio gerarchico tende a divenire meno conveniente. In riferimento all'analogia sulle metodologie di programmazione vista in precedenza, il programmatore tende a personalizzare le routine software quando la velocità di esecuzione è cruciale; i compilatori (o gli strumenti di progettazione) non sono ancora in grado di sostituire l'uomo.

Materiale protetto da copyright

- È importante, inoltre, notare che l'approccio basato sull'astrazione è corretto solo fino ad un certo punto. Le prestazioni di un sommatore, per esempio, possono essere pesantemente influenzate dal modo in cui questo è collegato agli altri blocchi del sistema. Le stesse interconnessioni contribuiscono a ritardare il segnale, poiché introducono capacità, resistenze e perfino induttanze parassite. L'impatto degli elementi parassiti dovuti alle interconnessioni è destinato a peggiorare negli anni a venire con l'evoluzione della tecnologia e la conseguente riduzione delle dimensioni geometriche dei dispositivi *(scaling)*.
- Lo scaling tecnologico tende a enfatizzare altre carenze del modello basato sull'astrazione. Alcuni elementi del sistema tendono ad essere *globali* o *esterni*. Esempi di fattori globali sono i segnali di *clock*, usati per sincronizzare un sistema digitale, e le linee di alimentazione. L'aumento delle dimensioni di un sistema digitale influenza drammaticamente tali segnali globali. Connettere un numero maggiore di celle ad una linea di alimentazione, per esempio, può causare una caduta di tensione lungo la pista che, a sua volta, può rallentare tutte le celle connesse alla stessa linea. Problematiche quali la distribuzione dei segnali dei clock, la sincronizzazione e la distribuzione della tensione di alimentazione stanno diventando sempre più critiche. Affrontare queste problematiche richiede una comprensione approfondita della complessità di un circuito digitale.
- Un altro effetto dell'evoluzione tecnologica è che nel corso del tempo emergono continuamente nuovi vincoli e problematiche progettuali. Un tipico esempio è il continuo riemergere del consumo di potenza come fattore limitante, come si è visto nella panorama storica. Un altro esempio è il cambiamento del rapporto tra i parametri parassiti dei dispositivi e quelli delle interconnessioni. Per affrontare questi fattori imprevedibili, è necessario essere almeno in grado di modellare e analizzare il loro impatto, cosa che richiede una comprensione approfondita della topologia e del comportamento del circuito.
- Per concludere, citando Murphy: quando qualcosa può andare male, spesso lo fa. Il circuito fabbricato non è sempre conforme a quanto previsto dalle simulazioni, anche se accurate. Piccole deviazioni sono sempre possibili a causa delle variazioni statistiche dei parametri del processo di fabbricazione, o a causa delle induttanze del package, o a causa di segnali di clock non correttamente modellati. Risolvere tali problemi in un circuito richiede esperienza progettuale.

Per tutte queste ragioni, siamo convinti che una conoscenza approfondita delle tecniche e degli approcci progettuali sia un patrimonio essenziale per un progettista di sistemi digitali. Sebbene questi potrebbe non aver a che fare quotidianamente con i dettagli circuitali, queste conoscenze gli saranno d'aiuto per gestire eventuali problemi inattesi e per determinare quali siano gli effetti dominanti durante l'analisi di un sistema.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

caso in cui il segnale di clock del secondo registro è ritardato (sfasato) di una quantità δ. Il fronte di salita del clock sfasato ϕ' ritarderà il campionamento del segnale di ingresso nel secondo registro. Se il tempo necessario alla propagazione dall'uscita del primo registro all'ingresso del secondo registro è inferiore al ritardo del clock, il secondo registro campionerà un valore errato. Ciò causa un cambiamento prematuro del segnale di uscita, come è chiaramente mostrato nella simulazione, dove il segnale *Out'* assume valore alto in corrispondenza del primo fronte di salita di ϕ' anziché del secondo. Nei termini della nostra analogia con il semaforo, le vetture che attraversano il primo semaforo tamponerebbero le vetture al semaforo successivo che non sono ancora partite.

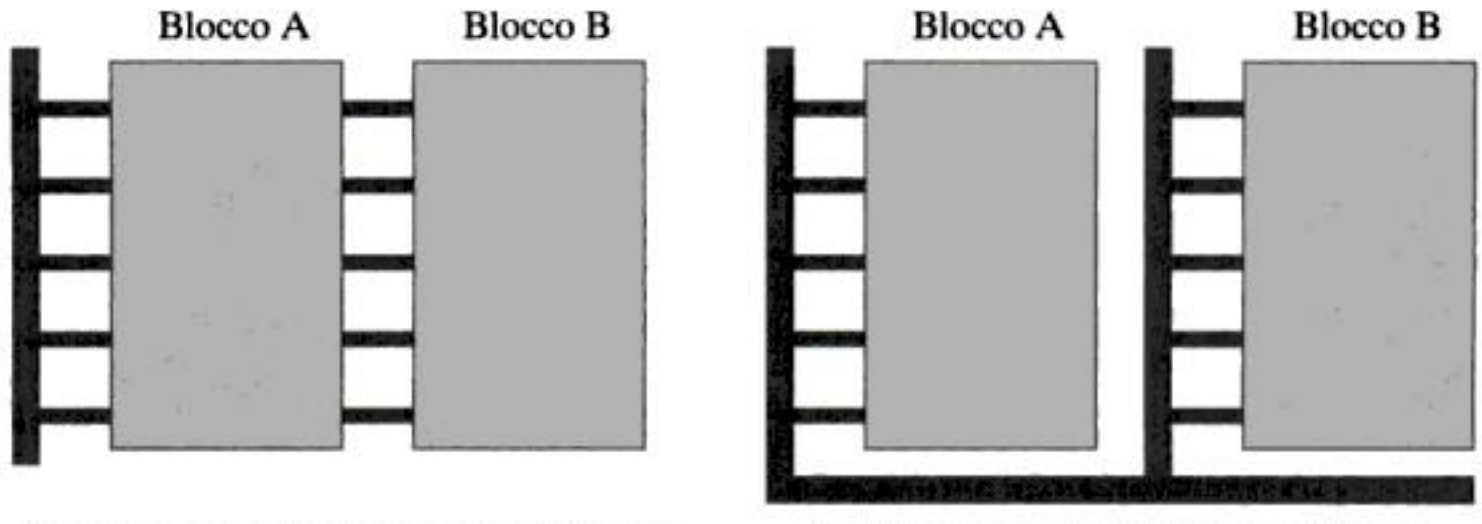
Il disallineamento del clock, o *clock skew*, è un esempio significativo di come segnali globali posssano compromettere il funzionamento di un sistema progettato a livelli gerarchici. In effetti, il *clock skew* è uno dei problemi progettuali più critici che si presentano ai progettisti di sistemi complessi e dalle elevate prestazioni.

Esempio 1.2 Struttura gerarchica e la distribuzione dell'alimentazione

Se il segnale di clock è un tipico esempio di segnale globale che va al di là dei confini della struttura gerarchica, la rete di distribuzione dell'alimentazione ne rappresenta un secondo. Un sistema digitale richiede una tensione di alimentazione continua e stabile che deve essere portata a ciascuna porta logica. Per assicurare il corretto funzionamento, la tensione deve essere stabile entro poche decine di millivolt, quindi il sistema di distribuzione dell'alimentazione deve poter garantire la stabilità anche in presenza di ampie variazioni di corrente. La natura resistiva delle connessioni all'interno del chip e le induttanze associate ai piedini del *package* fanno sì che ciò sia molto difficile da garantire. Si pensi che la corrente media assorbita da un processore che consuma una potenza di 100 W ed è alimentato a 1V è uguale a 100 A. Il picco massimo di corrente può essere facilmente il doppio e il valore istantaneo della corrente assorbita dall'alimentazione può variare bruscamente da zero al valore di picco in brevissimo tempo (nell'ordine di 1 nsec o meno). Ciò comporta variazioni di corrente di 100 GA/s, valore sorprendentemente elevato.

Consideriamo il problema della resistenza delle linee di distribuzione. Per una corrente di 100 A e una resistenza dell'interconnessione di 1.25 mΩ otteniamo una diminuzione del 5% della tensione di alimentazione (con un'alimentazione di 2.5 V). Facendo le interconnessioni più larghe la resistenza diminuisce e, quindi, anche la caduta di tensione. Il dimensionamento della rete di distribuzione dell'alimentazione è relativamente semplice da un punto di vista meramente circuitale, ma è di gran lunga più complesso in un sistema progettato in modo gerarchico. Prendiamo, per esempio, consideriamo i due blocchi disegnati in Figura 1-8a [Saleh01]. Se si analizzasse e ottimizzasse la distribuzione dell'alimentazione all'interno del blocco *A* separatamente dal resto del circuito, non si terrebbe conto del carico addizionale dovuto al blocco *B*. Se il

Materiale protetto da copyright



(a) Interconnessioni attraverso il blocco (b)Interconnessioni attorno al blocco

Figura 1-8 Linee di distribuzione della tensione di alimentazione.

percorso che porta l'alimentazione al blocco *B* passasse attraverso il blocco *A*, ci sarebbe una sensibile caduta di tensione all'ingresso del blocco *B*, poiché parte della potenza erogata sarebbe assorbita dal blocco *A* prima di raggiungere il blocco *B*.

Poiché tale caduta di tensione deriva dalla resistenza dell'interconnessione tra i due blocchi, le linee di interconnessione dovrebbero essere fatte girare attorno al blocco *A* distribuendo l'alimentazione a ciascun blocco mediante linee separate, come illustrato in Figura 1-8b. Idealmente, il ramo principale deve essere sufficientemente largo per sopportare l'intero flusso di corrente assorbito complessivamente dai rami secondari. Sebbene la distribuzione dell'alimentazione in questo modo sia abbastanza semplice da gestire e controllare, la sua realizzazione richiede un maggiore impegno di area. La pista metallica che costituisce il ramo principale deve essere dimensionata per condurre la corrente di entrambi i blocchi. Questa esigenza costringe i progettisti ad assegnare un area aggiuntiva riservata alle linee di alimentazione secondarie che si staccano dalla linea principale.

Quando si aggiungono sempre più blocchi al sistema, le complesse interazioni tra un blocco e l'altro determinano l'effettiva caduta di tensione. Non è sempre possibile prevedere esattamente quale percorso seguirà la corrente se esistono più cammini conduttivi dall'alimentazione verso le varie porte o blocchi logici attivi. Inoltre, la corrente assorbita dai vari moduli raggiunge raramente il valore di picco nel medesimo istante. Tutte queste considerazioni rendono la progettazione delle linee di distribuzione dell'alimentazione un compito arduo. Ciò richiede un metodo di progettazione e un approccio che vada al di là dei limiti imposti dalla rigida procedura gerarchica.

Lo scopo di questo libro è quello di fornire un ponte tra la visione astratta di un sistema digitale e il circuito digitale che esso rappresenta con le sue peculiarità. Partendo dalla profonda comprensione del funzionamento dei dispositivi elettronici e dall'analisi dettagliata del nucleo fondamentale di un sistema digitale (l'invertitore) sposteremo gradualmente queste conoscenze verso la progettazione di strutture più complesse: porte logiche, registri, blocchi aritmetici, e memorie. Nel progettare ciascuno dei moduli sopra citati, il progettista è continuamente posto davanti al problema di identificarne i parametri dominati, individuare la parte

Materiale protetto da copyright

critica del sistema da ottimizzare e determinare le proprietà specifiche che rendono il modulo considerato (per esempio una memoria) diverso da qualunque altro. Nei capitoli seguenti analizzeremo altre questioni globali di grande interesse nella moderna progettazione digitale, come il *consumo di potenza*, le *interconnessioni*, la *temporizzazione* e il *sincronismo*.

1.3 Figure di merito di un sistema digitale

Questo paragrafo introduce e definisce le principali *figure di merito* dei sistemi digitali. Queste metriche aiutano a quantificare la qualità di un sistema digitale da diversi punti di vista: costi, funzionalità, robustezza, prestazioni, e consumo di energia. Quale tra queste caratteristiche sia la più importante dipende dal tipo di applicazione. Per esempio, la pura e semplice velocità è una caratteristica fondamentale per un server. D'altro canto, se consideriamo le applicazioni portatili, per esempio un telefono cellulare, il consumo di energia è la figura di merito dominante. Le proprietà appena introdotte sono importanti a qualsiasi livello di astrazione: sistema, chip, modulo o porta logica. Per assicurare la coerenza delle definizioni lungo tutta la struttura gerarchica del sistema, useremo il cosiddetto approccio *bottom-up*: inizieremo definendo le figure di merito di un semplice invertitore e procederemo, attraverso i vari livelli di astrazione, verso strutture progressivamente più complesse: porta logica, modulo e sistema.

1.3.1 Costo di un circuito integrato

Il costo complessivo di qualsiasi prodotto può essere suddiviso in due componenti: i *costi variabili* e i *costi fissi*.

Costi fissi

I costi fissi sono indipendenti dal numero di pezzi venduti. Un contributo importante ai costi fissi di un circuito integrato sono le risorse in termini di forza lavorativa che occorre a produrre il sistema. Questo costo è fortemente influenzato dalla complessità del sistema, dalla rigidità delle caratteristiche richieste e dalla produttività dei progettisti. Le tecniche avanzate di progettazione che permettono di gestire in modo automatico la maggior parte dei passi del processo possono contribuire ad aumentare la produttività. Tenere bassi i costi di progettazione in presenza di circuiti integrati con complessità sempre crescente è una delle grandi sfide a cui è deve rispondere l'industria dei semiconduttori.

È necessario, inoltre, considerare i *costi indiretti*, ossia tutti i costi aggiuntivi che una ditta deve sostenere, ma che non può imputare direttamente ad uno specifico prodotto. Essi includono, fra tanti: i costi di ricerca e sviluppo (*R&D, Research and Development*), l'apparato di fabbricazione, i costi di pubblicità e vendita, e il mantenimento delle infrastrutture.

Costi variabili

I costi variabili tengono conto di tutto ciò che è attribuibile direttamente al singolo prodotto e, quindi, sono proporzionali al volume della produzione. I costi variabili comprendono il costo delle parti utilizzate nel prodotto, i costi di assemblaggio e i costi di collaudo. Il costo totale di un circuito integrato (IC) è dato dalla seguente equazione:

Materiale protetto da copyright

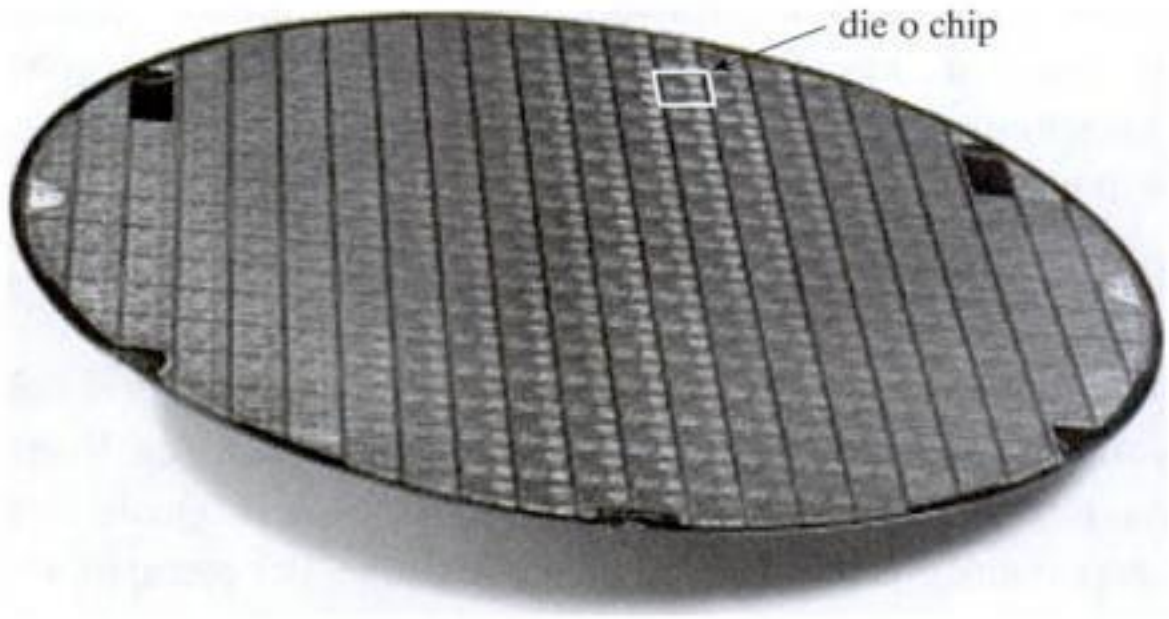


Figura 1-9 Wafer finito. Ciascun rettangolo rappresenta un *chip* — in questo caso il microprocessore AMD Duron™ (riprodotto con il permesso di AMD).

$$\text{costo per IC} = \text{costo variabile per IC} + \left(\frac{\text{costo fisso}}{\text{volume}}\right) \tag{1.1}$$

L'impatto dei costi fissi è maggiore nei prodotti con piccoli volumi di produzione. Ciò spiega perché ha senso avere grandi gruppi di progettisti che lavorano per diversi anni su prodotti di largo successo come i microprocessori.

Mentre il costo di produzione di un singolo transistor è calato esponenzialmente negli ultimi decenni, l'equazione che descrive i costi variabili non è cambiata affatto:

$$\text{costi variabili} = \frac{\text{costo chip} + \text{costo test} + \text{costo packaging}}{\text{resa finale}} \tag{1.2}$$

Come si vedrà più avanti nel Capitolo 2, il processo di fabbricazione dei circuiti integrati raggruppa un certo numero di circuiti identici all'interno di una singola fetta di silicio (o *wafer*) (vedi Figura 1-9). Alla fine del processo di fabbricazione, il wafer è tagliato in *chip* o *die*, (letteralmente: *quadratini* o *schegge*) che, dopo la fase di test, vengono montati individualmente nel contenitore o *package*). In questo paragrafo ci concentreremo al costo di un chip. Il costo del test e del *packaging* sarà oggetto dei capitoli seguenti.

Il costo di un chip dipende dal numero di chip per ciascun wafer, e dalla percentuale di questi che sono completamente funzionanti. Questo secondo fattore è detto *resa*. Il costo di un chip è dato dalla formula:

$$\text{costo chip} = \frac{\text{costo wafer}}{\text{num. chip per wafer} \times \text{resa}} \tag{1.3}$$

Il numero di chip per wafer è essenzialmente l'area del wafer divisa per l'area di un chip. In realtà, la situazione è un po' più complessa, poiché i wafer sono circolari e i chip sono rettangolari. Quindi, i chip lungo il perimetro del wafer sono sostanzialmente perduti. La dimensione del wafer sta costantemente aumentando nel corso degli anni, permettendo di fabbricare sempre più chip per ogni lotto di produzione. L'Equazione (1.3) indica inoltre che

Materiale protetto da copyright

il costo di un circuito dipende anche dalla sua area; aumentare l'area del chip significa dunque ridurre il numero di chip per wafer.

L'effettiva relazione tra costo e area è ancor più complessa e dipende dalla resa del processo. Sia i materiali impiegati sia il processo di fabbricazione introducono difetti e imperfezioni che possono portare al malfunzionamento del chip. Assumendo che i difetti siano uniformemente distribuiti sulla superficie del wafer e la resa sia inversamente proporzionale alla complessità del processo di fabbricazione, otteniamo l'equazione seguente:

$$\text{resa} = \left(1 + \frac{\text{difetti per unità di area} \times \text{area del chip}}{\alpha}\right)^{-\alpha} \tag{1.4}$$

dove α è un parametro che dipende dalla complessità del processo di fabbricazione, ed è circa proporzionale al numero di maschere utilizzate. Una buona stima per gli attuali processi CMOS più complessi è $\alpha = 3$. Il numero di difetti per unità di area è una misura delle imperfezioni dei materiali e del processo. Oggigiorno, una densità tra 0.5 e 1 difetti/cm^2 è abbastanza tipica, anche se tale valore dipende molto da quanto è "consolidato" il processo.

Esempio 1.3 Resa

Consideriamo un wafer con diametro 30 cm (= 12 pollici), un chip con area 2.5 cm^2, una densità di difetti pari a 1 difetto/cm^2 e $\alpha = 3$. Calcoliamo la resa di un questo processo CMOS.

Il numero di chip per wafer può essere stimato con la seguente relazione, che tiene conto dei chip perduti nella regione perimetrale del wafer:

$$\text{chip per wafer} = \frac{\pi \times (\text{diametro wafer}/2)^2}{\text{area chip}} - \frac{\pi \times \text{diametro wafer}}{\sqrt{2 \times \text{area chip}}}$$

Ciò significa che ci sono 252 (= 296 – 44) chip potenzialmente funzionanti, in questo particolare esempio. La resa del processo, calcolata mediante l'Equazione (1.4), è pari al 16%. Ciò significa che, in media, solo 40 chip (su 252) saranno completamente funzionanti.

Il concetto di base è che il numero di chip completamente funzionanti in ciascun wafer, quindi il costo per chip, dipende fortemente dall'area del chip stesso. Mentre la resa tende ad essere molto buona per sistemi molto piccoli, questa cala rapidamente non appena il sistema raggiunge una certa complessità. Tenendo presenti le relazioni introdotte in precedenza e i valori tipici dei parametri in gioco, possiamo concludere che il costo per chip è proporzionale alla quarta potenza dell'area:

$$\text{costo per chip} = f(\text{area chip}^4) \tag{1.5}$$

L'area è un parametro controllato direttamente dai progettisti, ed è la figura di merito primaria del costo. Anche per una singola porta logica è auspicabile un'area più piccola possibile.

Materiale protetto da copyright

Minore è l'area della porta, maggiore sarà il livello di integrazione e, quindi, minore sarà l'area complessiva del chip. Inoltre, una porta logica piccola tende ad essere più veloce e consumare meno; la capacità parassita complessiva associata ad una data porta, che è uno dei parametri dominanti per le prestazioni di un circuito, spesso è proporzionale all'area.

Il *numero di transistor* in una porta logica è indicativo dell'occupazione di area prevista, ma anche altri parametri possono avere un certo peso. Ad esempio, uno schema troppo complesso di connessioni fra i transistor potrebbe portare la capacità parassita delle piste di interconnessione a essere dominante. Anche la *complessità di una porta*, espressa come il numero di transistor, e la regolarità nella struttura delle interconnessioni, giocano il loro ruolo nel costo di realizzazione di un progetto. Le strutture complesse sono molto difficili da costruire e tendono a portar via molto tempo utile ai progettisti. Semplicità e regolarità sono caratteristiche molto preziose per un progetto efficiente dal punto di vista dei costi.

1.3.2 Funzionalità e robustezza

Uno dei requisiti primari di un circuito digitale è, ovviamente, che svolga la funzione per cui è stato progettato. Normalmente, il comportamento di un circuito misurato all'uscita differisce da quello nominale atteso. Una delle ragioni di questa deviazione sta nella dispersione statistica dei parametri del processo di fabbricazione. Le dimensioni geometriche e i parametri dei dispositivi variano tra lotti di produzione diversi o anche all'interno dello stesso wafer o chip. Il comportamento elettrico di un circuito può risentire profondamente di queste variazioni. La presenza di sorgenti di rumore e disturbi dentro e fuori il chip è un ulteriore fonte di variazioni della risposta di un circuito. La parola rumore (*noise*), nel contesto dei circuiti digitali, significa *variazione indesiderata della tensione e della corrente di un nodo logico.* Rumore e disturbi possono influenzare il circuito in molti modi. Alcuni esempi di sorgenti di rumore digitale sono illustrati dalla Figura 1-10. Due linee collocate l'una accanto all'altra in un circuito integrato, per esempio, danno luogo ad accoppiamenti capacitivi e mutue induttanze. Quindi, una variazione nella corrente o nella tensione su una delle due linee può interferire con il segnale nella linea vicina. I disturbi nella rete di distribuzione della tensione di alimentazione o nelle linee di massa possono influenzare i livelli di tensione all'uscita di una porta logica.

La maggior parte dei disturbi in un sistema digitale provengono dal suo interno, e l'ampiezza di tali disturbi è proporzionale all'escursione dei segnali digitali (*swing*). L'inter-

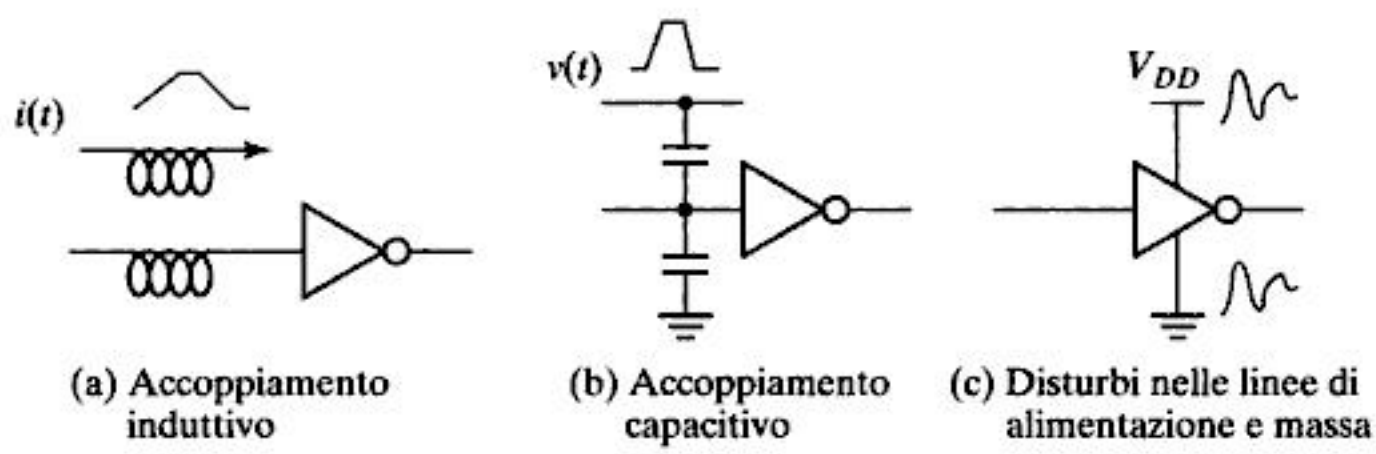


Figura 1-10 Sorgenti di rumore nei circuiti digitali.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

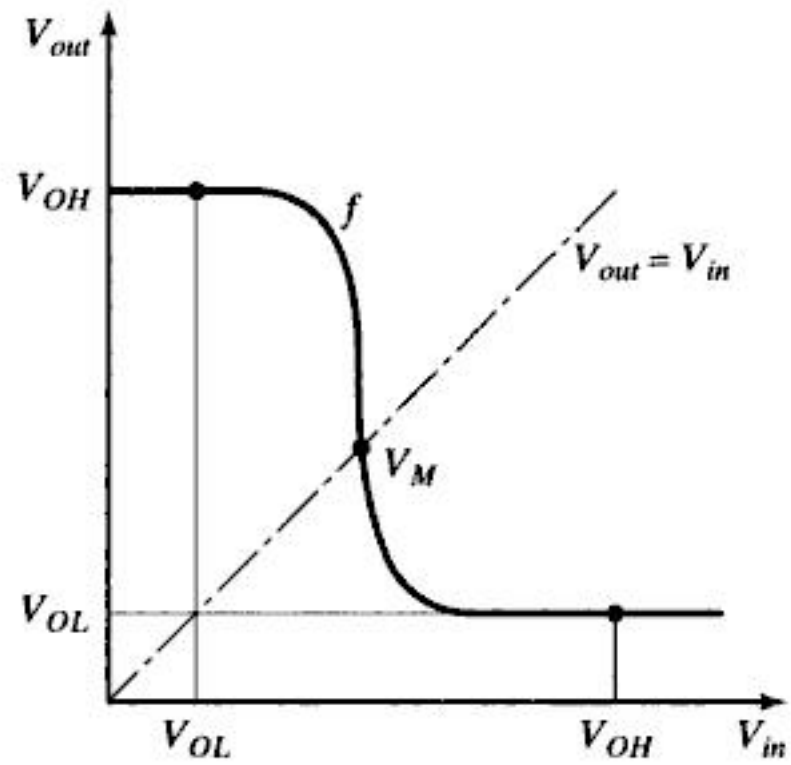
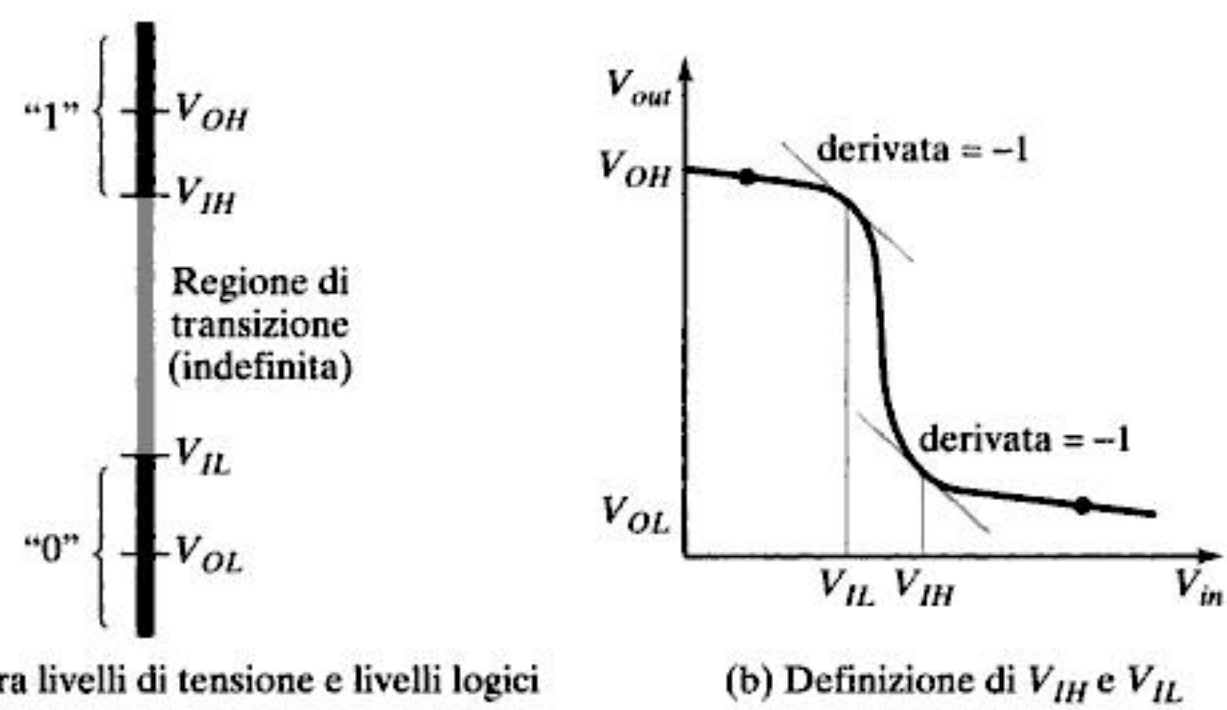


Figura 1-11 Caratteristica di trasferimento ingresso-uscita di un invertitore.

è illustrato in Figura 1-11. I valori nominali delle tensioni associate al livello logico alto e basso, V_{OH} e V_{OL}, possono essere facilmente determinati: $V_{OH} = f(V_{OL})$ e $V_{OL} = f(V_{OH})$. Un altro punto di interesse nella VTC è la *tensione di soglia logica* o *tensione di commutazione* (*switching threshold voltage*) V_M, definita dall'equazione $V_M = f(V_M)$ (V_M non va confusa con la tensione di soglia dei transistor). V_M si può determinare in due modi: graficamente, come l'intersezione della VTC con la retta $V_{out} = V_{in}$; oppure elettricamente, cortocircuitando l'uscita con l'ingresso e calcolando la tensione di uscita all'equilibrio. La tensione di soglia logica rappresenta il punto intermedio dell'escursione e si dimostra di particolare interesse quando si studiano i circuiti con retroazione (detti anche *circuiti sequenziali*).

Anche se all'ingresso di una porta logica fosse applicato il valore di tensione ideale, il segnale di uscita spesso può essere diverso dal valore nominalmente atteso. Tale deviazione può essere conseguenza di disturbi oppure effetto del carico connesso all'uscita della porta (ad esempio, dal numero di porte logiche connesse all'uscita). La Figura 1-12a illustra il



(a) Relazione tra livelli di tensione e livelli logici (b) Definizione di V_{IH} e V_{IL}

Figura 1-12 Associazione dei segnali logici ai livelli di tensione.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

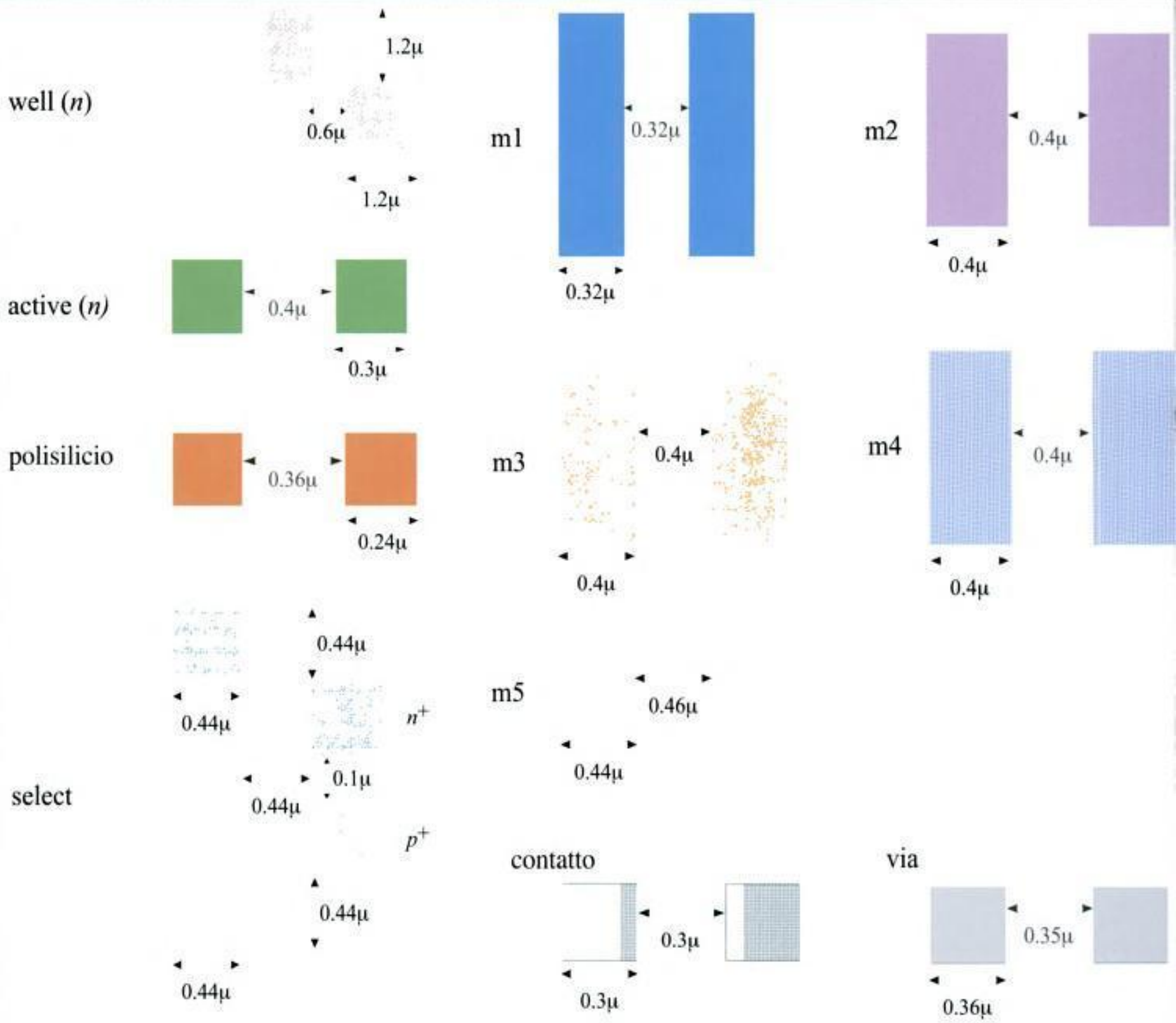


Tavola 2. Regole di layout intralayer: dimensioni e distanze minime

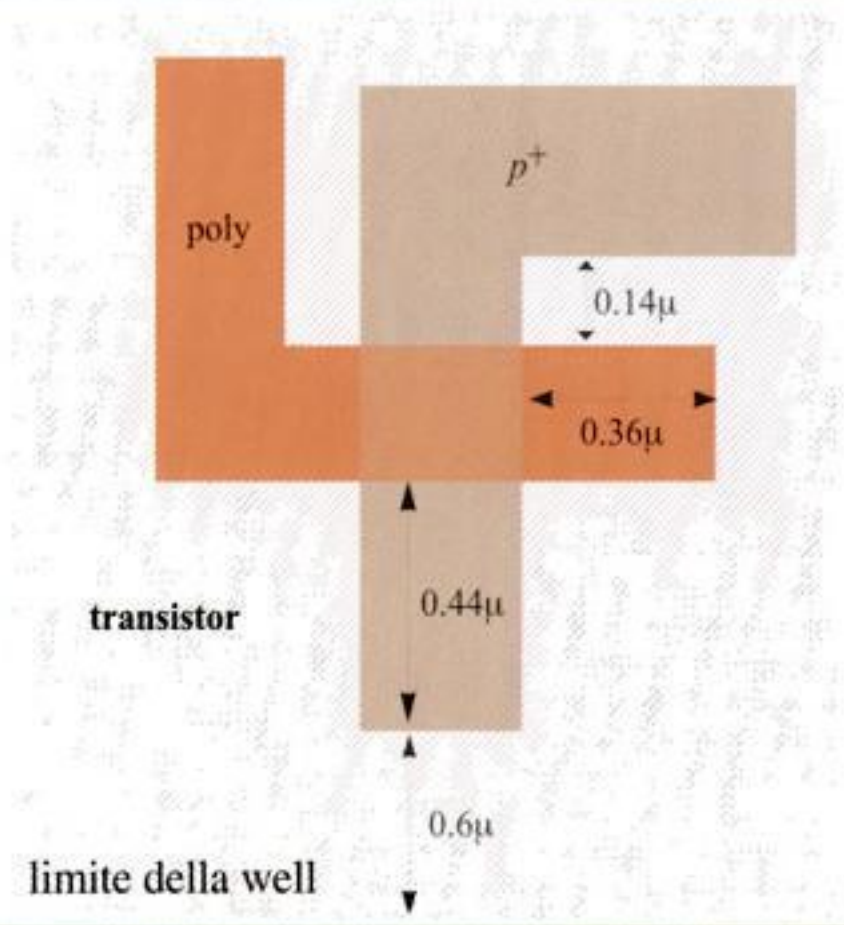


Tavola 3. Regole di layout per un transistor PMOS

Materiale protetto da copyright

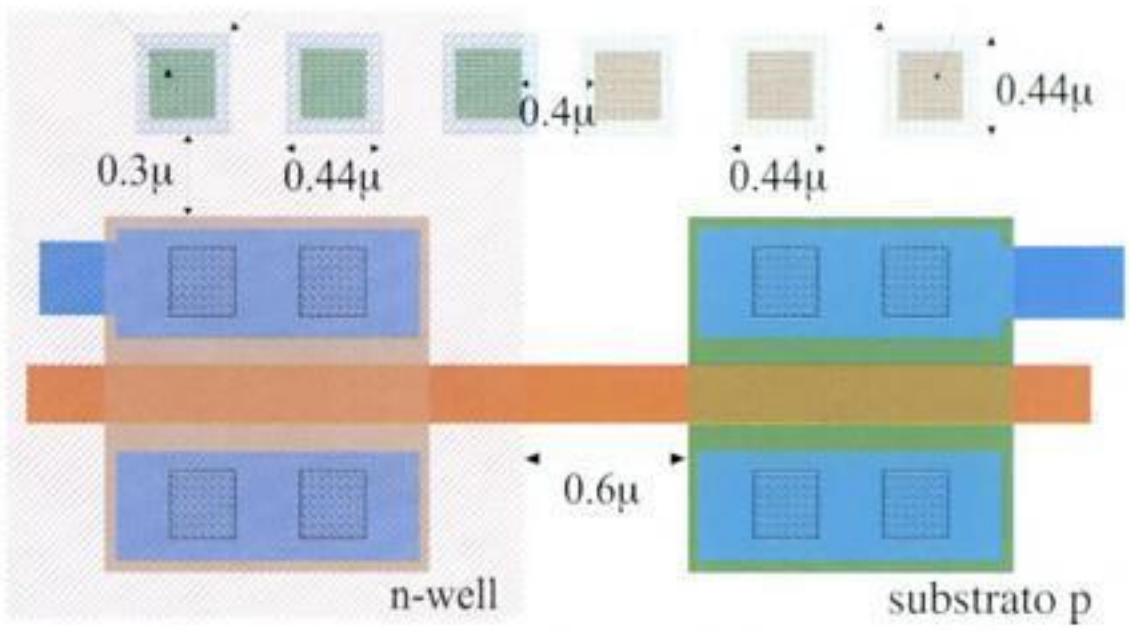


Tavola 5. Regole di layout per contatti di well e layer select

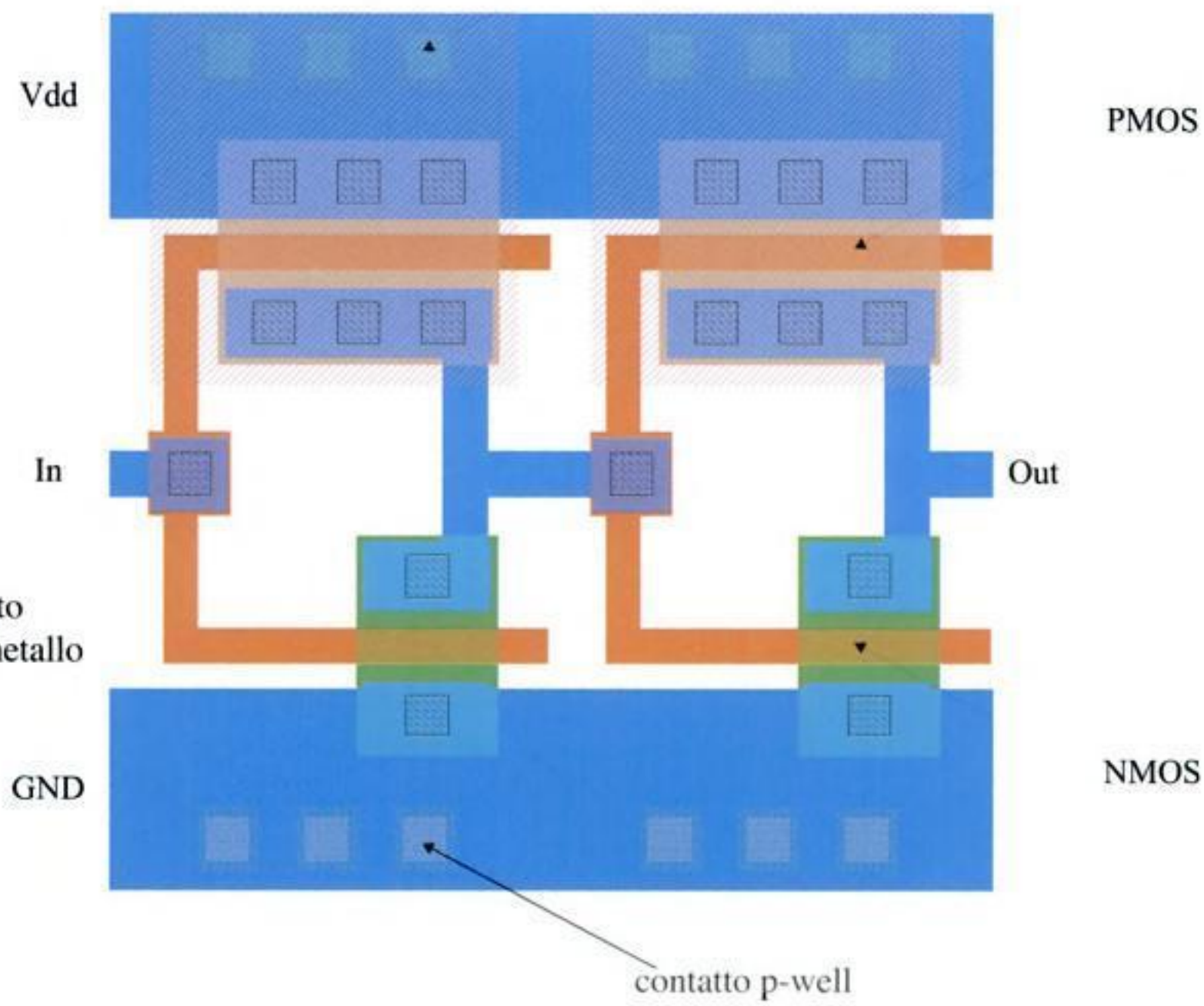


Tavola 6. Layout di un invertitore nella tecnologia CMOS da 0.25 μm

Materiale protetto da copyright

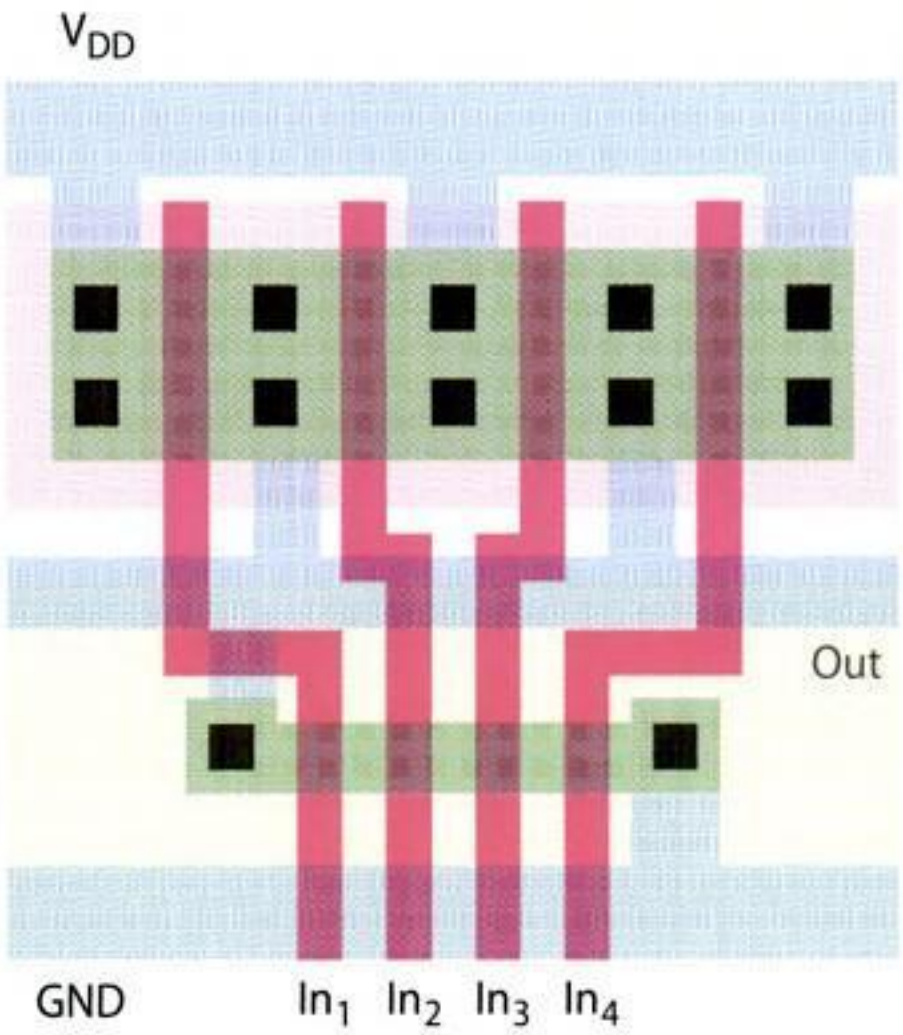


Tavola 7. Layout di una porta NAND CMOS a quattro ingressi

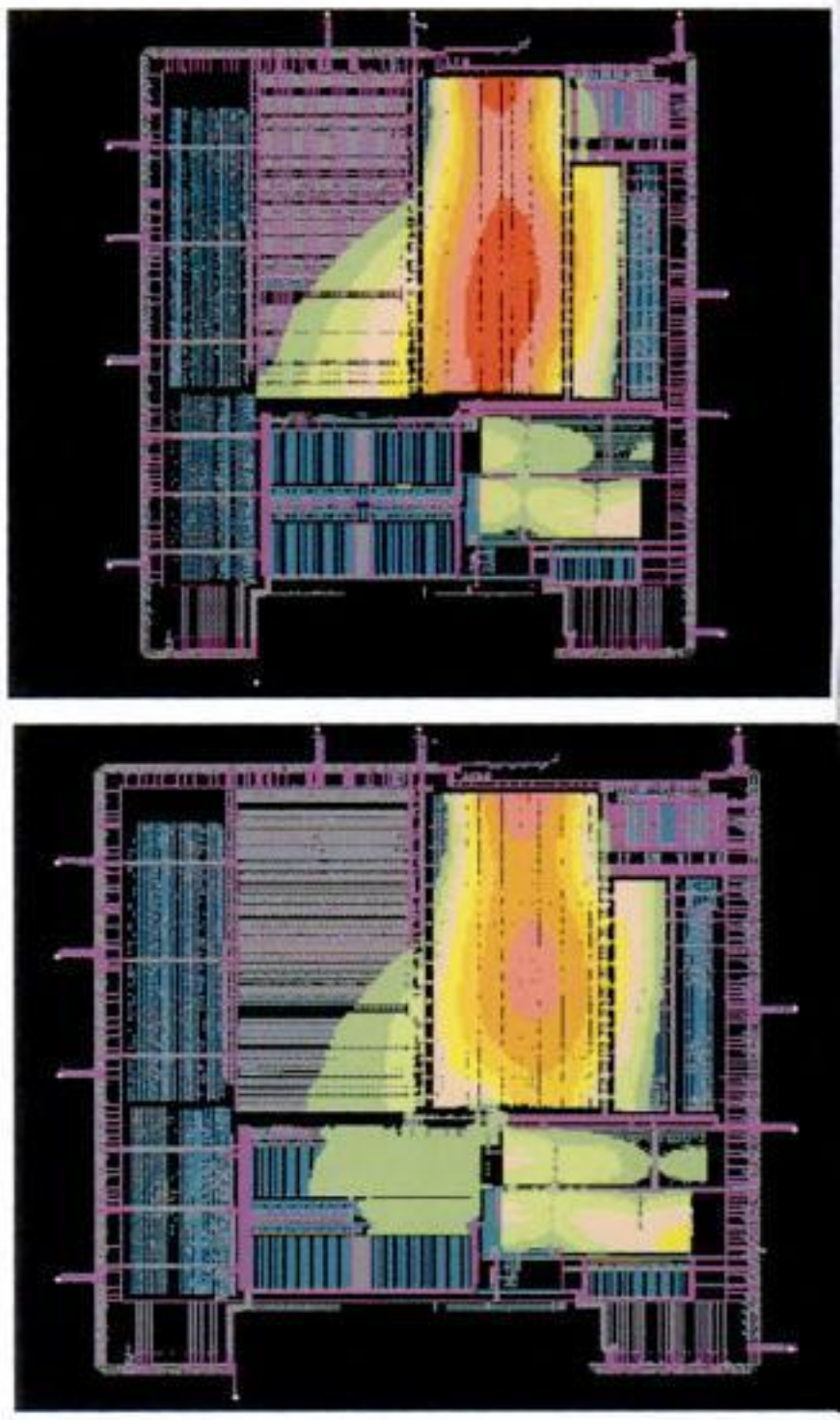
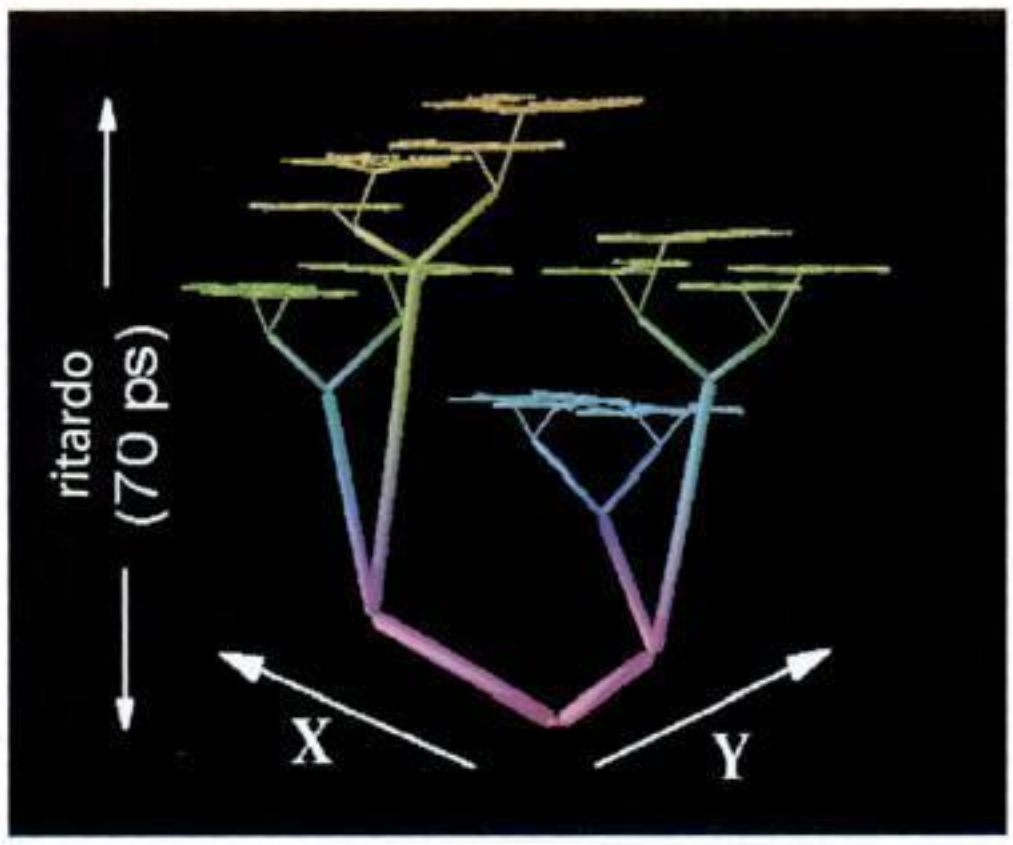


Tavola 8. Caduta di tensione IR simulata in una rete di distribuzione dell'alimentazion



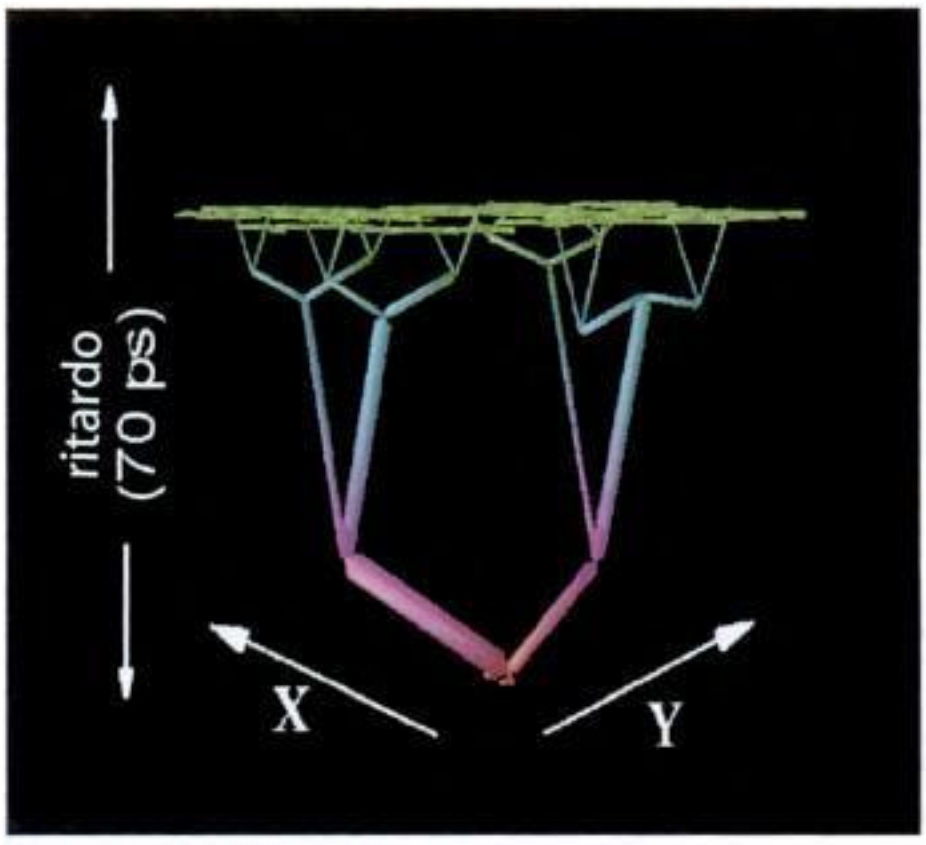


Tavola 9. Visualizzazione del ritardo del clock in una rete ad albero che pilota carichi diversi

Materiale protetto da copyright

modo in cui un dato valore logico è rappresentato nella realtà da un intervallo continuo di valori di tensione permessi, separati da una regione di valore indefinito, piuttosto che da un solo valore nominale. Le regioni permesse definiscono i valori logici alto e basso e sono delimitate dai livelli di tensione V_{IH} e V_{IL}, rispettivamente. Per definizione, questi rappresentano i punti dove il guadagno della VTC (dV_{out} / dV_{in}) è uguale a -1 come schematizzato in Figura 1-12b. La regione compresa tra V_{IH} e V_{IL} è detta *regione indefinita* (talvolta chiamata anche *regione di transizione*). Per garantire il corretto funzionamento del circuito è necessario evitare che i valori a regime dei segnali di ingresso siano all'interno di questa regione.

Margini di rumore

Affinché una porta logica sia robusta e insensibile ai disturbi, è essenziale che l'intervallo di tensione che separa le regioni associate ai valori "0" e "1" sia il più ampio possibile. Una misura della sensibilità di una porta logica ai disturbi è data dai *margini di rumore* NM_L (*noise margin low*) e NM_H (*noise margin high*), che quantificano l'ampiezza delle regioni permesse "0" e "1" e definiscono il valore di soglia massimo consentito all'eventuale disturbo:

$$\begin{aligned} NM_L &= V_{IL} - V_{OL} \\ NM_H &= V_{OH} - V_{IH} \end{aligned} \tag{1.8}$$

I margini di rumore rappresentano il massimo livello di rumore che può essere tollerato all'ingresso quando le porte logiche sono connesse in cascata, com'è illustrato in Figura 1-13. Risulta chiaro che, affinché un circuito digitale funzioni correttamente, i margini di immunità al rumore devono essere maggiori di zero e, preferibilmente, più ampi possibile.

Proprietà rigenerativa

Ottenere ampi margini di rumore è uno degli obiettivi primari, ma certamente non è l'unico requisito. Supponiamo che un segnale sia affetto da un certo livello di rumore e differisca dal suo valore nominale di tensione. Fino a che il segnale resta all'interno dei margini di rumore,

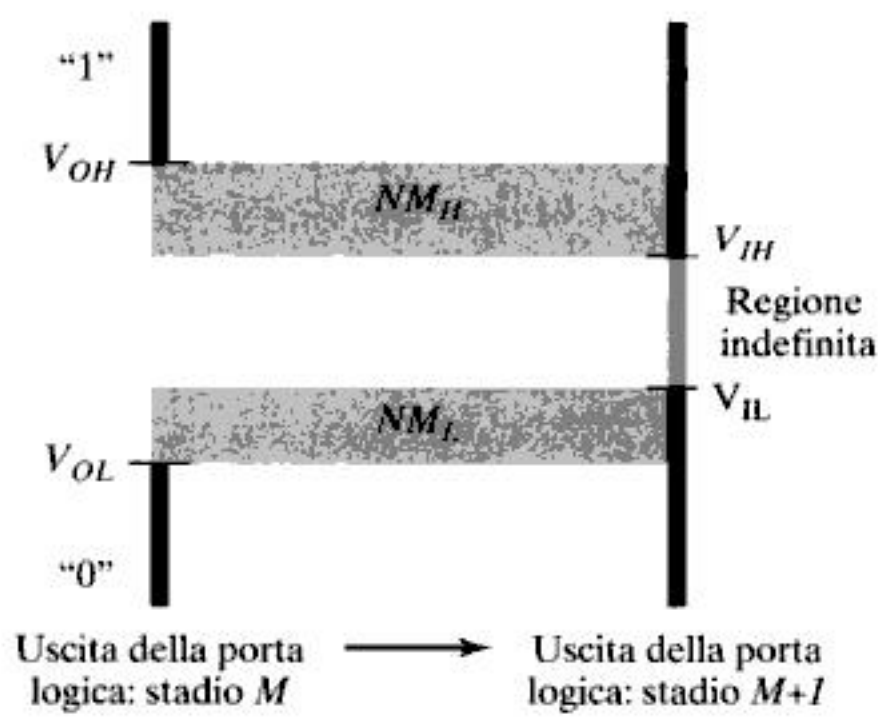
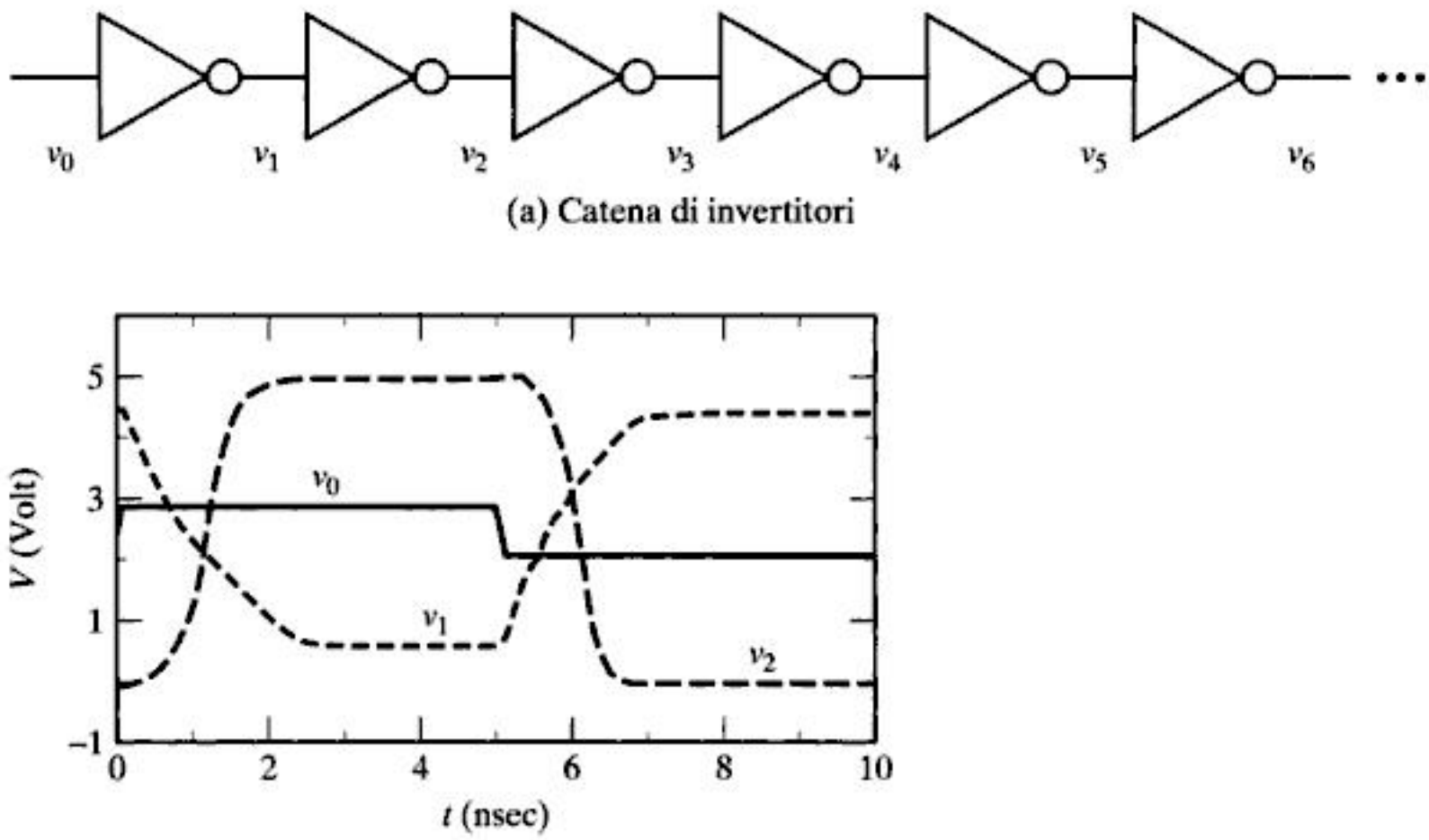


Figura 1-13 Cascata di porte invertenti: definizione dei margini di rumore.

Materiale protetto da copyright



(a) Catena di invertitori

(b) Simulazione della risposta di una catena di invertitori CMOS

Figura 1-14 La proprietà rigenerativa.

la porta logica che segue continua a funzionare correttamente, sebbene il valore di tensione del nodo di uscita sia diverso da quello nominalmente atteso. Questa variazione deve essere sommata a tutti gli altri disturbi che interessano il nodo di uscita; il segnale così distorto è infine trasmesso all'ingresso della porta successiva. Gli effetti di diverse sorgenti di rumore possono, quindi, sommarsi ed eventualmente forzare il segnale di uscita all'interno dell'intervallo logico indefinito. Fortunatamente, ciò non accade se la porta possiede la cosiddetta *proprietà rigenerativa*, che assicura che l'eventuale disturbo presente in un nodo converga gradualmente ad uno dei due livelli nominali e ideali di tensione dopo aver attraversato un certo numero di stadi logici. Questa proprietà può essere riformulata nel modo seguente:

Sia v_{in} ($v_{in} \in$ "0") la tensione applicata all'ingresso di una catena di N invertitori (Figura 1-14a). Assumendo che il numero di invertitori della catena sia pari, la tensione di uscita v_{out} (per N che tende a ∞) sarà uguale a V_{OL} se e solo se l'invertitore possiede la proprietà rigenerativa. Analogamente, quando all'ingresso della catena è applicata una tensione v_{in} associata al valore logico alto ($v_{in} \in$ "1"), la tensione di uscita convergerà al valore nominale V_{OH}.

Esempio 1.4 Proprietà rigenerativa

Il concetto di rigenerazione è illustrato nella figura Figura 1-14b, che riporta il comportamento simulato di una catena di invertitori CMOS. Il segnale di ingresso alla catena è un'onda quadra ideale con ampiezza ridotta a causa del rumore. Anziché commutare tra la massa e la tensione di alimentazione (5 V in questo esempio), il segnale di ingresso ha un'escursione limitata tra 2.1 e 2.9 V. Dalla simulazione si può osservare che tale degradazione del segnale tende rapidamente a scomparire man mano che il

Materiale protetto da copyright

segnale si propaga attraverso la catena; v_1, ad esempio, ha un'escursione da 0.6 V a 4.45 V. Allo stadio successivo, v_2 ha già un piena escursione tra i due valori nominali V_{OL} e V_{OH}. Gli invertitori utilizzati in questo esempio possiedono inequivocabilmente la proprietà rigenerativa.

La condizione sotto la quale una porta logica è rigenerativa si può dedurre intuitivamente analizzando un caso molto semplice. Nella Figura 1-15a sono riportate la VTC di un invertitore $V_{out} = f(V_{in})$ e la sua funzione inversa *finv*( ), ottenuta semplicemente scambiando gli assi x e y e definita nel modo seguente:

$$in = f(out) \Rightarrow in = finv(out) \tag{1.9}$$

Assumiamo che un certo valore di tensione v_0, che differisce dal livello nominale, sia applicato all'ingresso del primo invertitore della catena. La tensione di uscita di tale invertitore sarà $v_1 = f(v_0)$ e sarà presente all'ingresso dell'invertitore successivo. Graficamente, ciò corrisponde a $v_1 = finv(v_2)$. Il segnale gradualmente converge al valore di tensione nominale dopo un certo numero di stadi invertenti, come indicato dalle frecce. In Figura 1-15b il segnale non converge verso nessuno dei due livelli nominali, ma si assesta a un valore di tensione intermedio. Quindi, la caratteristica di quest'ultima è *non rigenerativa*. La differenza tra questi due casi sta nel diverso guadagno della caratteristica di trasferimento delle due porte. Per essere rigenerativa, la VTC deve avere una regione di transizione (con valore logico indefinito) caratterizzata da un guadagno, in modulo, *maggiore di* 1, compresa tra due regioni permesse (con valore logico definito), dove il guadagno deve essere *minore di* 1. Una siffatta porta possiede due punti di lavoro stabili. Ciò chiarisce la definizione dei livelli di tensione V_{IH} e V_{IL} che sono esattamente i punti di separazione tra le regioni permesse (di valore logico definito) e la regione di transizione indefinita.

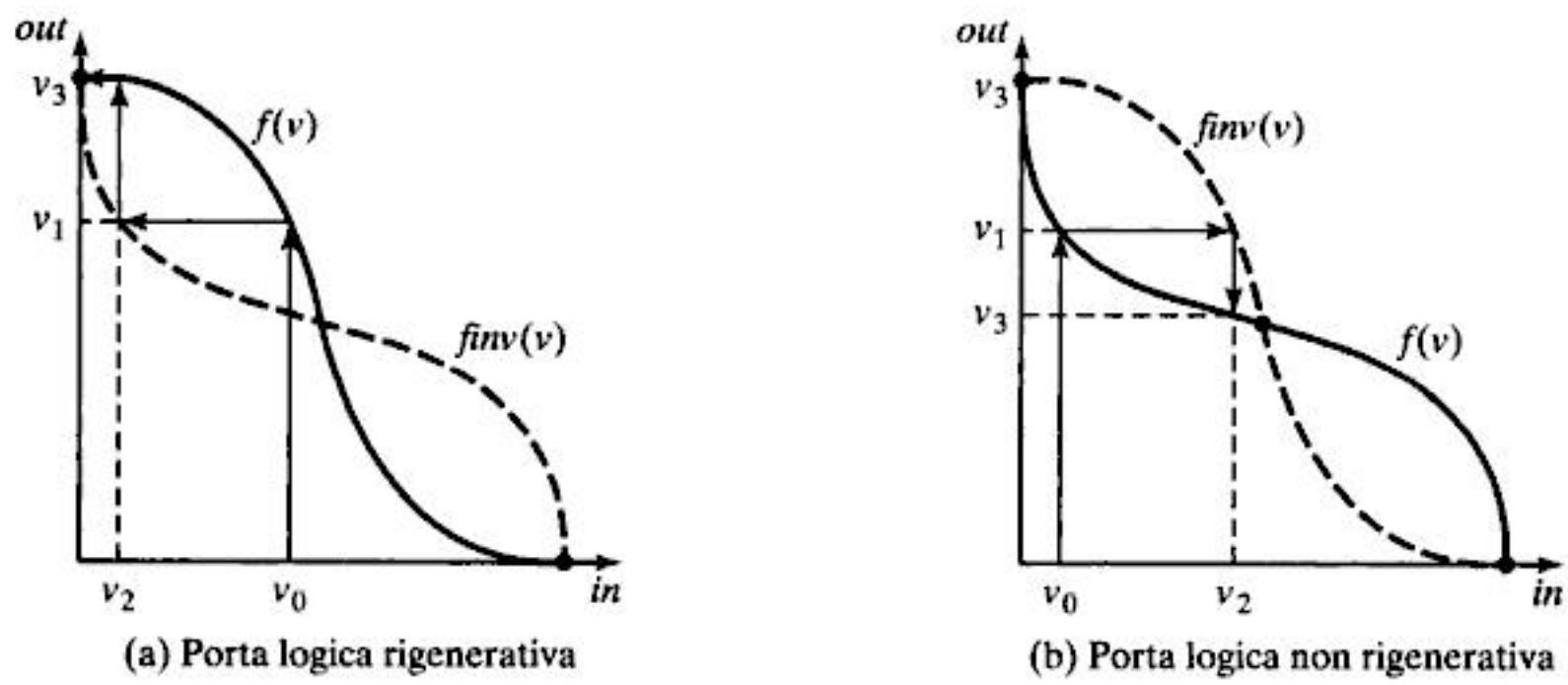


Figura 1-15 Condizione per la rigenerazione del segnale.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

valori del guadagno, aumentare ulteriormente l'escursione del segnale non dà alcun beneficio, poiché il rumore aumenta proporzionalmente all'escursione stessa. Nei prossimi capitoli, discuteremo di alcune famiglie logiche che, eliminando la maggior parte delle sorgenti interne di disturbi, possono lavorare con margini di rumore ed escursione di segnale molto ridotti.

Direzionalità

La caratteristica di direzionalità richiede a una porta logica di essere *unidirezionale;* ciò significa che qualsiasi variazione del livello di tensione in uscita di una porta non deve causare alcuna interferenza negli ingressi costanti dello stesso circuito. Se così non fosse, una commutazione del segnale di uscita sarebbe riflessa verso gli ingressi e sarebbe interpretata come un disturbo con effetti negativi sull'integrità del segnale.

Nella realizzazione pratica delle porte logiche, la completa unidirezionalità non può essere mai soddisfatta. Non si può eliminare totalmente la retroazione del segnale dall'uscita all'ingresso: gli accoppiamenti capacitivi tra gli ingressi e le uscite sono un esempio tipico di tale retroazione. Quindi è essenziale minimizzare tali accoppiamenti in modo tale da non compromettere il valore logico del segnale di ingresso.

Fan-in e fan-out

Il *fan-out* di una porta logica definisce *il numero N di porte logiche connesse all'uscita della porta in questione* (vedi Figura 1-16). Aumentare il fan-out di una porta può avere ripercussioni sul valore logico dell'uscita. Dal mondo degli amplificatori analogici, sappiamo che questo effetto può essere ridotto aumentando la resistenza di ingresso delle porte di carico (ridurre l'assorbimento di corrente all'ingresso) e mantenendo bassa la resistenza di uscita della porta che le pilota (riducendo gli effetti delle correnti di carico sulla tensione di uscita). Quando il fan-out è troppo grande, i carichi aggiunti in uscita possono deteriorare il comportamento dinamico della porta stessa. Per questi motivi, in molti componenti all'interno di librerie standard è definito un *fan-out massimo* per garantire che il comportamento statico e dinamico sia entro le specifiche fornite.

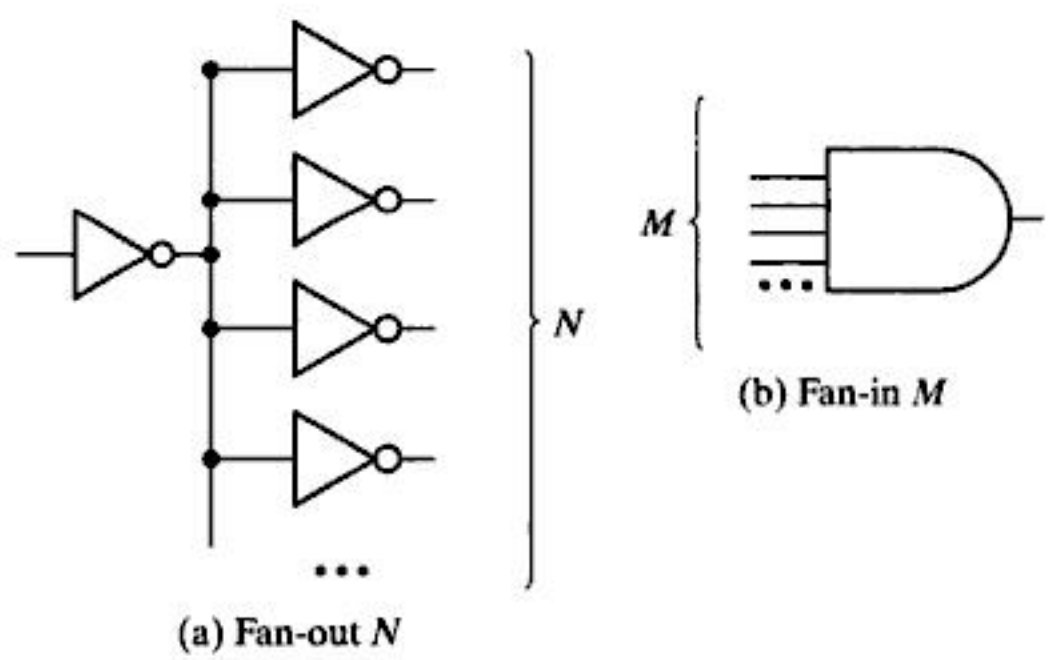


Figura 1-16 Definizione di fan-out e fan-in di una porta logica.

Materiale protetto da copyright

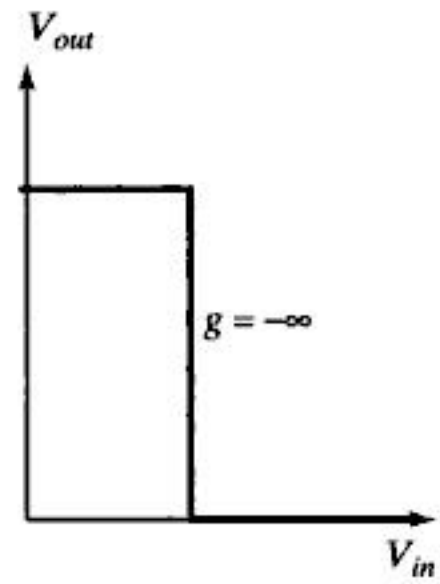


Figura 1-17 Caratteristica statica di trasferimento ideale.

Il *fan-in* di una porta è definito come il *numero di ingressi* della porta stessa (vedi Figura 1-16b). Le porte logiche con elevato fan-in tendono ad essere molto complesse; ciò comporta spesso minori prestazioni statiche e dinamiche.

La porta logica ideale

A partire da tutte queste osservazioni, possiamo definire la porta logica *ideale* dal punto di vista statico. Il modello ideale dell'invertitore è molto importante perché ci dà una misura di riferimento per valutare l'effettiva qualità delle porte logiche reali.

La VTC ideale è illustrata in Figura 1-17 e ha: guadagno infinito nella regione di transizione; tensione di soglia logica localizzata esattamente a metà dell'escursione logica; margini di rumore identici e uguali a metà dell'escursione logica; impedenza di ingresso infinita; e impedenza di uscita nulla (ovvero, la porta ha un fan-out illimitato). Sebbene questa VTC ideale sia impossibile da realizzare nella pratica, alcune porte, come l'invertitore CMOS statico, hanno caratteristiche molto vicine a quelle ideali.

Esempio 1.5 Caratteristica di trasferimento statica

La Figura 1-18 raffigura un esempio di caratteristica di trasferimento ingresso-uscita di un invertitore reale, anche se obsoleto. I valori dei parametri statici sono calcolati graficamente tramite la curva.

$$V_{OH} = 3.5\ \text{V}; \qquad V_{OL} = 0.45\ \text{V}$$

$$V_{IH} = 2.35\ \text{V}; \qquad V_{IL} = 0.66\ \text{V}$$

$$V_M = 1.64\ \text{V}$$

$$NM_H = 1.15\ \text{V}; \qquad NM_L = 0.21\ \text{V}$$

La caratteristica di trasferimento qui considerata è ovviamente ben lontana dall'idealità: è asimmetrica, ha un valore molto piccolo di NM_L e l'escursione logica è di soli 3.05 V, sostanzialmente minore del massimo valore ottenibile, 5 V (che è il valore della tensione di alimentazione per questa particolare tecnologia).

Materiale protetto da copyright

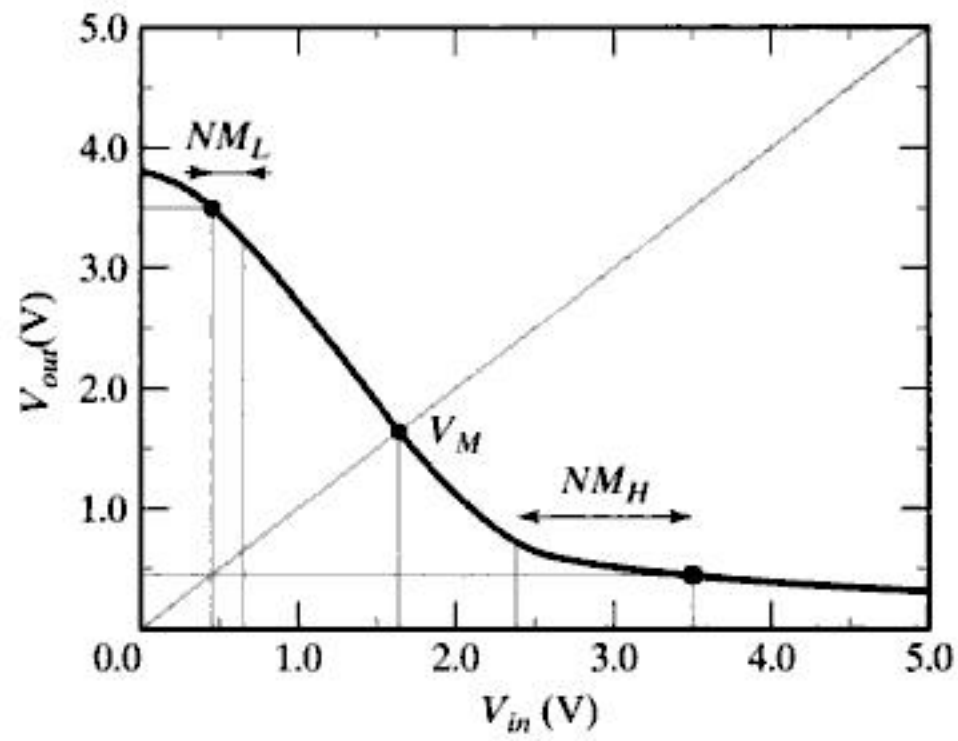


Figura 1-18 VTC di un invertitore NMOS degli anni '70.

1.3.3 Prestazioni

Dal punto di vista di chi progetta il sistema, le prestazioni di un circuito esprimono la sua capacità computazionale. Un microprocessore, per esempio, è spesso caratterizzato dal numero di istruzioni al secondo che è in grado di eseguire. Questa figura di merito delle prestazioni dipende sia dall'architettura del microprocessore (ad esempio, il numero di operazioni che può eseguire in parallelo) sia dalla struttura dei circuiti logici. Sebbene la prima ricopra un ruolo cruciale, non è trattata in questo testo; il lettore può far riferimento a molti eccellenti libri su questo argomento [ad esempio, Hennessy02]. Quando si guarda al sistema da un punto di vista puramente progettuale, le prestazioni sono spesso misurate dalla durata del periodo del clock (*tempo di ciclo*), o dal numero di cicli al secondo (*frequenza di clock*). Per una data tecnologia e un dato sistema, il valore minimo del periodo di clock è determinato da un certo numero di fattori, tra i quali, il tempo che i segnali impiegano a propagarsi attraverso gli stadi logici, il tempo necessario a leggere e scrivere le informazioni nei registri e le incertezze legate al tempo di propagazione, e quindi di arrivo, dei segnali di clock. Ciascuno di questi dettagli sarà discusso ampiamente in questo libro. Comunque, le prestazioni di una data porta logica, presa individualmente, stanno al centro dell'analisi delle prestazioni dell'intero sistema.

Il *tempo di propagazione* (o *tempo di ritardo*) t_p di una porta logica è indice di quanto velocemente essa risponde a un cambiamento della configurazione dei suoi ingressi. Esso misura *il ritardo che il segnale subisce quando attraversa la porta logica*. Come è illustrato in Figura 1-19 nel caso di una porta invertente, il tempo di propagazione è misurato dall'istante in cui il segnale di ingresso arriva al 50% della sua escursione logica all'istante in cui l'uscita raggiunge il 50% della propria transizione.[2] Poiché una porta logica risponde con tempi di propagazione diversi secondo il tipo di commutazione dell'ingresso, sono necessarie due defini-

[2]La convenzione di considerare il 50% come livello di riferimento è motivata dall'assunzione che la tensione di soglia logica (soglia di commutazione) V_M sia tipicamente localizzata a metà dell'escursione logica.

Materiale protetto da copyright

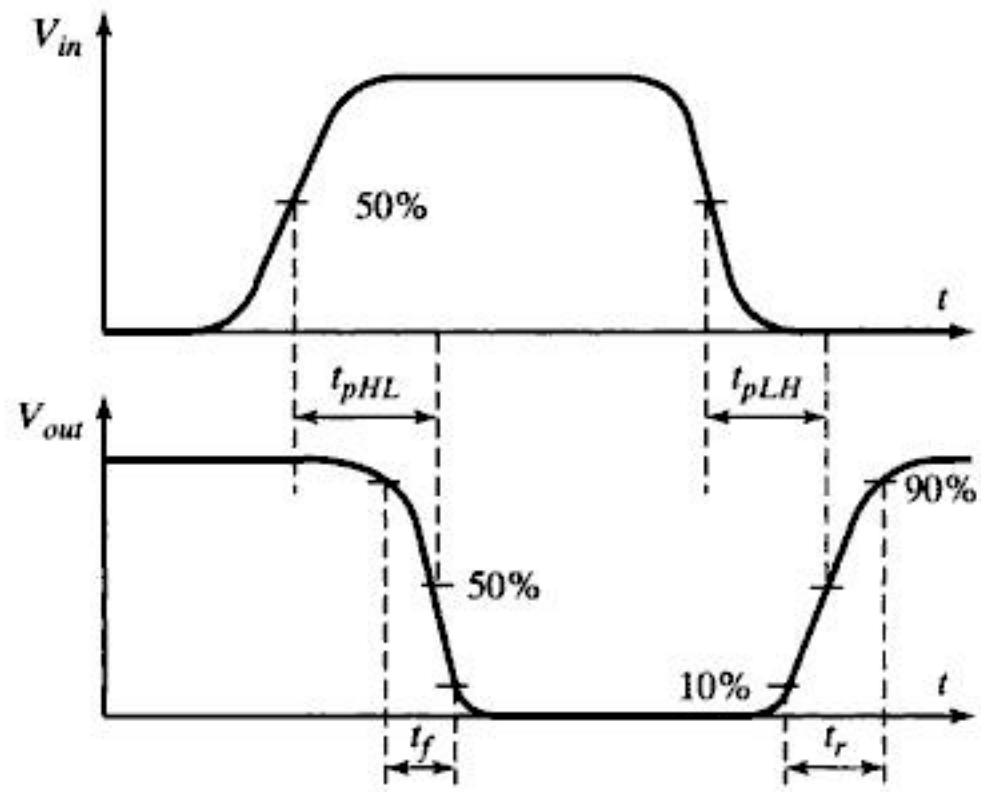


Figura 1-19 Definizione di tempo di propagazione, tempo di salita e tempo di discesa.

zioni del tempo di propagazione. Con t_{pLH} si definisce il tempo di propagazione di una porta logica quando l'uscita esegue una commutazione positiva, o basso-alto (*low-to-high*), mentre con t_{pHL} si fa riferimento ad una commutazione negativa, o alto-basso (*high-to-low*) del segnale di uscita. Il tempo di propagazione o tempo di ritardo t_p è per convenzione definito come la media dei due:

$$t_p = \frac{t_{pLH} + t_{pHL}}{2} \tag{1.12}$$

ATTENZIONE. Vale la pena notare che il tempo di propagazione t_p, contrariamente a t_{pLH} e t_{pHL}, è una figura di merito convenzionale, ma in sé priva di significato fisico. Essa è usato soprattutto per confrontare differenti tecnologie, circuiti o famiglie logiche.

Il tempo di propagazione non è funzione solo della tecnologia e topologia del circuito, ma dipende anche da altri fattori, tra i quali il più importante è la velocità di commutazione dei segnali di ingresso e di uscita. Per misurare queste grandezze, introduciamo i tempi di *salita* (*rise time*) e *discesa* (*fall time*), t_r e t_f, che sono caratteristici delle forme d'onda piuttosto che della porta logica (vedi Figura 1-19), ed esprimono la velocità con cui un segnale commuta tra i due livelli. Per tener conto dell'incertezza sui precisi istanti di inizio e fine della commutazione, i tempi di salita e discesa sono definiti tra il 10% e il 90% dell'escursione del segnale, com'è illustrato in figura. I tempi di salita e discesa di un segnale sono strettamente legati alla resistenza di uscita della porta e alla capacità di carico connessa al nodo di uscita.

Quando si confrontano le prestazioni di porte logiche realizzate con tecnologie e stili di progettazione diversi, è importante non creare confusione introducendo parametri esterni come gli effetti di carico, il fan-in e il fan-out. In questi casi sarebbe opportuno avere un

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Esempio 1.6 Tempo di propagazione di una rete *RC* del primo ordine

I circuiti digitali sono spesso rappresentati con reti *RC* del primo ordine come quello in Figura 1-21. Il tempo di propagazione di tale circuito è quindi di notevole interesse.

Quando all'ingresso del circuito si applica un gradino di tensione (con v_{in} che passa da 0 a V), la risposta è esponenziale nel tempo ed è data dalla seguente espressione (dove $\tau = RC$, la costante di tempo della rete):

$$v_{out}(t) = (1 - e^{-t/\tau})\, V \tag{1.13}$$

Il tempo richiesto a raggiungere il 50% dell'intera escursione è $t = \ln(2)\tau = 0.69\tau$. Analogamente, il tempo necessario a commutare dal 10% al 90% è $t = \ln(9)\tau = 2.2\tau$. Vale la pena di ricordare questi numeri perché saranno usati spesso in questo testo.

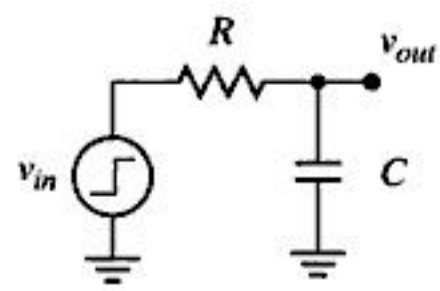


Figura 1-21 Rete *RC* del primo ordine.

1.3.4 Consumo di potenza ed energia

Il consumo di potenza di un sistema definisce quanta energia il circuito assorbe per ciascuna operazione e, quindi, quanto calore dissipa il circuito stesso. Questo valore influenza molte delle scelte progettuali, come la massima corrente erogabile dalla tensione di alimentazione, la durata della batteria, la dimensione delle linee di alimentazione, i requisiti del package e del sistema di raffreddamento. Dunque, la dissipazione di potenza è una caratteristica fondamentale del sistema che ne determina la fattibilità, i costi e l'affidabilità. Nel mondo dell'elaborazione ad alte prestazioni il limite al consumo di potenza, imposto dal package e dal sistema di dissipazione del calore, determina il numero di porte che si possono integrare in un singolo chip e la massima velocità di commutazione ammissibile. Con la crescente diffusione delle applicazioni portatili, la quantità di energia a disposizione è limitata dall'impiego di batterie e pone un limite superiore al numero di operazioni che possono essere svolte.

A seconda del tipo di problema progettuale che si presenta, si deve considerare una figura di merito diversa. Ad esempio, la potenza di picco P_{max} è importante nel dimensionamento delle linee di alimentazione. Quando si considerano il raffreddamento o i requisiti della batteria, ha maggior interesse il consumo medio di potenza P_{media}. I due parametri sono definiti come segue:

$$\begin{aligned} P_{max} &= i_{DD,\,max} V_{DD} = max[p(t)] \\ P_{media} &= \frac{1}{T}\int_0^T p(t)dt = \frac{V_{DD}}{T}\int_0^T i_{DD}(t)dt \end{aligned} \tag{1.14}$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

$$E_C = \int_0^{\infty} i_C(t) v_{out}(t) dt = \int_0^{\infty} C \frac{dv_{out}}{dt} v_{out} dt = C \int_0^{V} v_{out} dv_{out} = \frac{CV^2}{2} \tag{1.16}$$

Questa è esattamente la metà dell'energia sottratta alla sorgente del segnale. Per chi si chiedesse che cosa accade all'altra metà dell'energia assorbita — una semplice analisi mostrerebbe che tale quantità viene dissipata sotto forma di calore nella resistenza durante la carica. Lasciamo dimostrare al lettore il fatto che, durante la fase di scarica (con il gradino inverso della tensione V_{in} da V a 0), l'energia originariamente immagazzinata nella capacità viene dissipata nella resistenza e convertita in calore.

1.4 Riepilogo

In questo capitolo introduttivo abbiamo studiato la storia e l'evoluzione dei circuiti e sistemi digitali. Abbiamo inoltre introdotto le figure di merito impiegate per valutare la qualità di un sistema: costo, funzionalità, robustezza, prestazioni e dissipazione di potenza (ed energia).

1.5 Approfondimenti

La progettazione di circuiti integrati digitali è stata ed è tuttora oggetto di molti testi e monografie. Per aiutare il lettore a trovare ulteriori informazioni su alcuni argomenti, riportiamo di seguito un ampio elenco di riferimenti bibliografici. Lo stato dell'arte e gli sviluppi nell'area dei sistemi digitali è generalmente riportato in riviste tecniche o negli atti di conferenze, dei quali i più importanti sono elencati di seguito.

Riviste e atti di conferenze

IEEE Journal of Solid-State Circuits
IEICE Transactions on Electronics (Japan)
Proceedings of The International Solid-State and Circuits Conference (ISSCC)
Proceedings of the VLSI Circuits Symposium
Proceedings of the Custom Integrated Circuits Conference (CICC)
European Solid-State Circuits Conference (ESSCIRC)

Bibliografia

MOS

M. Annaratone, *Digital CMOS Circuit Design*, Kluwer, 1986.
T. Dillinger, *VLSI Engineering*, Prentice Hall, 1988.
M. Elmasry, ed., *Digital MOS Integrated Circuits*, IEEE Press, 1981.
M. Elmasry, ed., *Digital MOS Integrated Circuits II*, IEEE Press, 1992.
L. Glasser and D. Dopperpuhl, *The Design and Analysis of VLSI Circuits*, Addison-Wesley, 1985.
A. Kang and Leblebici, *CMOS Digital Integrated Circuits*, 2nd Ed., McGraw-Hill, 1999.
C. Mead and L. Conway, *Introduction to VLSI Systems*, Addison-Wesley, 1980.
K. Martin, *Digital Integrated Circuit Design*, Oxford University Press, 2000.
D. Pucknell and K. Eshraghian, *Basic VLSI Design*, Prentice Hall, 1988.

Materiale protetto da copyright

M. Shoji, *CMOS Digital Circuit Technology*, Prentice Hall, 1988.
J. Uyemura, *Circuit Design for CMOS VLSI*, Kluwer, 1992.
H. Veendrick, Deep-Submicron *CMOS IC's: From Basics to ASICS*, 2nd Edition, Kluwer Academic Publishers, 2000.
N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, Addison-Wesley, 1985, 1993.

Progettazione di Sistemi ad alte prestazioni
K. Bernstein et al, *High Speed CMOS Design Styles*, Kluwer Academic, 1998.
A. Chandrakasan, F. Fox, and W. Bowhill, ed., *Design of High-Performance Microprocessor Circuits*, IEEE Press, 2000.
M. Shoji, *High-Speed Digital Circuits*, Addison-Wesley, 1996.

Progettazione di sistemi a basso consumo
A. Chandrakasan and R. Brodersen, ed., *Low-Power Digital CMOS Design*, IEEE Press, 1998.
M. Pedram and J. Rabaey, ed., *Power-Aware Design Methodologies*, Kluwer Academic, 2002.
J. Rabaey and M. Pedram, ed., *Low-Power Design Methodologies*, Kluwer Academic, 1996.
G. Yeap, *Practical Low-Power CMOS Design*, Kluwer Academic, 1998.

Memorie
K. Itoh, *VLSI Memory Chip Design*, Springer, 2001.
B. Keeth and R. Baker, *DRAM Circuit Design*, IEEE Press, 1999.
B. Prince, *Semiconductor Memories*, Wiley, 1991.
B. Prince, *High Performance Memories*, Wiley, 1996.
D. Hodges, *Semiconductor Memories*, IEEE Press, 1972.

Interconnessioni e packaging
H. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI*, Addison-Wesley, 1990.
W. Dally and J. Poulton, *Digital Systems Engineering*, Cambridge University Press, 1998.
E. Friedman, ed., *Clock Distribution Networks in VLSI Circuits and Systems*, IEEE Press, 1995.
J. Lau et al, ed., *Electronic Packaging: Design, Materials, Process, and Reliability*, McGraw-Hill, 1998.

Strumenti e metodologie di progetto
V. Agrawal and S. Seth, *Test Generation for VLSI Chips*, IEEE Press, 1988.
D. Clein, *CMOS IC Layout*, Newnes, 2000.
G. De Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, 1994.
S. Rubin, *Computer Aids for VLSI Design*, Addison-Wesley, 1987.
J. Uyemura, *Physical Design of CMOS Integrated Circuits Using L-Edit*, PWS, 1995.
A. Vladimirescu, *The Spice Book*, John Wiley and Sons, 1993.
W. Wolf, *Modern VLSI Design*, Prentice Hall, 1998.

Tecnologie bipolari e BiCMOS
A. Alvarez, *BiCMOS Technology and Its Applications*, Kluwer, 1989.
M. Elmasry, ed., *BiCMOS Integrated Circuit Design*, IEEE Press, 1994.
S. Embabi, A. Bellaouar, and M. Elmasry, *Digital BiCMOS Integrated Circuit Design*, Kluwer, 1993.

Varie
J. Buchanan, *CMOS/TTL Digital Systems Design*, McGraw-Hill, 1990.
H. Haznedar, *Digital Micro-Electronics*, Benjamin/Cummings, 1991.
D. Hodges and H. Jackson, *Analysis and Design of Digital Integrated Circuits*, 2nd ed., McGraw-Hill, 1988.
M. Smith, *Application-Specific Integrated Circuits*, Addison-Wesley, 1997.
R. K. Watts, *Submicron Integrated Circuits*, Wiley, 1989.

Materiale protetto da copyright

Riferimenti scientifici

[Bardeen48] J. Bardeen and W. Brattain, "The Transistor, a Semiconductor Triode," *Phys. Rev.*, vol. 74, p. 230, July 15, 1948.

[Beeson62] R. Beeson and H. Ruegg, "New Forms of All Transistor Logic," *ISSCC Digest of Technical Papers*, pp. 10–11, Feb. 1962.

[Dally98] B. Dally, *Digital Systems Engineering*, Cambridge University Press, 1998.

[Faggin72] F. Faggin, M.E. Hoff, Jr, H. Feeney, S. Mazor, M. Shima, "The MCS-4 - An LSI MicroComputer System," 1972 IEEE Region Six Conference Record, San Diego, CA, pp. 1–6, April 1972.

[Harris56] J. Harris, "Direct-Coupled Transistor Logic Circuitry in Digital Computers," *ISSCC Digest of Technical Papers*, p. 9, Feb. 1956.

[Hart72] C. Hart and M. Slob, "Integrated Injection Logic—A New Approach to LSI," *ISSCC Digest of Technical Papers*, pp. 92–93, Feb. 1972.

[Hoff70] E. Hoff, "Silicon-Gate Dynamic MOS Crams 1,024 Bits on a Chip," *Electronics*, pp. 68–73, August 3, 1970.

[Intel01] "Moore's Law", *http://www.intel.com/research/silicon/mooreslaw.htm*

[Masaki74] A. Masaki, Y. Harada and T. Chiba, "200-Gate ECL Master-Slice LSI," *ISSCC Digest of Technical Papers*, pp. 62–63, Feb. 1974.

[Moore65] G. Moore, "Cramming more Components into Integrated Circuits," Electronics, Vol. 38, Nr 8, April 1965.

[Murphy93] B. Murphy, "Perspectives on Logic and Microprocessors," *Commemorative Supplement to the Digest of Technical Papers, ISSCC*, pp. 49–51, San Francisco, 1993.

[Norman60] R. Norman, J. Last and I. Haas, "Solid-State Micrologic Elements," *ISSCC Digest of Technical Papers*, pp. 82–83, Feb. 1960.

[Hennessy02] J. Hennessy, D. Patterson, and David Goldberg, *Computer Architecture A Quantitative Approach*, Third Edition, Morgan Kaufmann Publishers, 2002.

[Saleh01] R. Saleh, M. Benoit, and P, McCrorie, "Power Distribution Planning", Simplex Solutions, *http://www.simplex.com/wt/sec.php?page_name=wp_powerplan*

[Schockley49] W. Schockley, "The Theory of pn Junctions in Semiconductors and pn-Junction Transistors," *BSTJ*, vol. 28, p. 435, 1949.

[Shima74] M. Shima, F. Faggin and S. Mazor, "An N-Channel, 8-bit Single-Chip Microprocessor," *ISSCC Digest of Technical Papers*, pp. 56–57, Feb. 1974.

[Swade93] D. Swade, "Redeeming Charles Babbage's Mechanical Computer," *Scientific American*, pp. 86–91, February 1993.

[Wanlass63] F. Wanlass, and C. Sah, "Nanowatt logic Using Field-Effect Metal-Oxide Semiconductor Triodes," *ISSCC Digest of Technical Papers*, pp. 32–32, Feb. 1963.

Esercizi

Consultare la pagina web **http://bwrc.eecs.berkeley.edu/IcBook** per una lista aggiornata di problemi ed esercizi. Rendendo disponibili gli esercizi in formato elettronico anziché stampati, siamo in grado di garantire il continuo aggiornamento e seguire la rapida evoluzione delle attuali tecnologie e dei metodi di progettazione dei circuiti integrati digitali.

Materiale protetto da copyright

CAPITOLO

2

Il processo di fabbricazione CMOS

Introduzione ai processi manifatturieri dei circuiti integrati CMOS

Regole di Layout

Packaging dei circuiti integrati

Evoluzione e prospettive future delle tecnologie dei circuiti integrati

2.1 Introduzione

2.2 Fabbricazione dei circuiti integrati CMOS
- 2.2.1 Il wafer di silicio
- 2.2.2 Fotolitografia
- 2.2.3 Principali passi di processo
- 2.2.4 Descrizione esemplificata di un processo CMOS

2.3 Regole di layout: il contratto tra il progettista e il costruttore

2.4 Il packaging dei circuiti integrati
- 2.4.1 Materiali dei package
- 2.4.2 Livelli di interconnessioni
- 2.4.3 Considerazioni termiche nel packaging

2.5 Le prospettive: evoluzione dei processi tecnologici
- 2.5.1 Nel breve periodo
- 2.5.2 Nel lungo periodo

2.6 Riepilogo

2.7 Approfondimenti

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

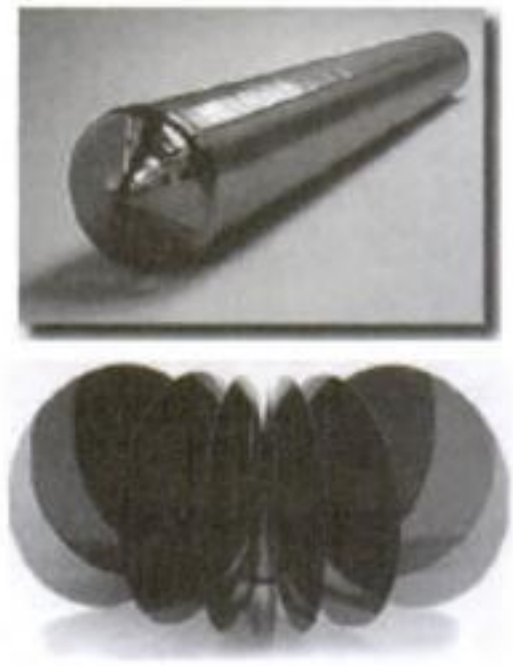


Figura 2-3 Lingotto di silicio monocristallino e wafer tagliati (da [Fullman99]).

2.2.2 Fotolitografia

In ogni passo di processo è necessario che un certa regione dell'area del wafer sia selezionata usando un'apposita maschera ottica, in modo che sia possibile eseguire le operazioni richieste solo su di essa, senza alterare le parti rimanenti. Ogni passo di processo consiste in una vasta gamma di operazioni: ossidazione, attacco chimico, deposizione di metallo o di polisilicio e impiantazione ionica. La tecnica impiegata per eseguire la selezione è detta *fotolitografia* ed è impiegata ripetutamente nel corso dell'intero processo. La Figura 2-4 illustra graficamente l'insieme delle diverse operazioni che sono coinvolte in un tipico ciclo fotolitografico. Possiamo identificare i seguenti passi principali:

1. *Ossidazione* — In questa fase (opzionale) un sottile strato di SiO_2 viene depositato sull'intera superficie del wafer esponendolo ad una miscela di ossigeno e idrogeno di elevata purezza e ad una temperatura di circa 1000°C. L'ossido può essere usato come strato isolante e può costituire anche il dielettrico di gate dei transistor.
2. *Stesura del fotoresist* — un polimero sensibile alla radiazione luminosa (simile al lattice) viene applicato sulla superficie del wafer, imponendo un movimento rotatorio alla fetta fino a formare una pellicola dello spessore di 1 μm circa. Questo composto, originariamente solubile in un solvente organico, se esposto alla radiazione luminosa, è in grado di formare forti legami tra i polimeri rendendo insolubili le regioni esposte. Un fotoresist (o resist) con questa caratteristica è definito *negativo.* Invece un fotoresist *positivo* ha la caratteristica opposta; è originariamente insolubile, ma diventa solubile dopo l'esposizione. Impiegando resist positivi o negativi la stessa maschera può essere usata per diversi passi di processo, selezionando due regioni complementari del wafer. Questo accorgimento è essenziale per diminuire il numero di maschere impiegate nel processo. Economicamente è un fattore prioritario, visto il crescente costo delle maschere dettato dall'evoluzione della tecnologia e dalla riduzione delle dimensioni dei dispositivi.

Materiale protetto da copyright

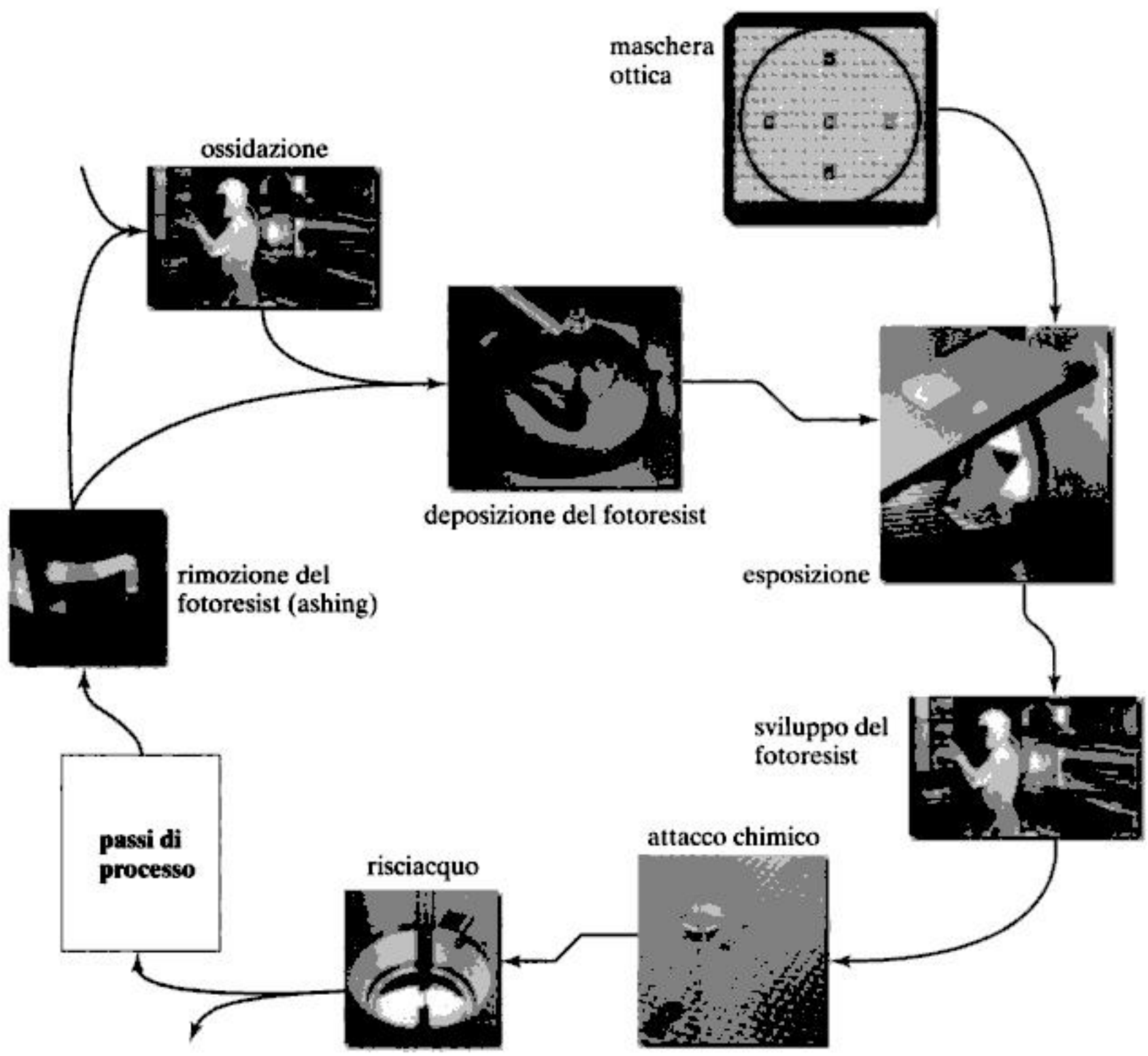


Figura 2-4 Operazioni svolte in un tipico ciclo fotolitografico (da [Fullman99]).

3. *Esposizione* — La maschera contenente l'insieme delle strutture che vogliamo trasferire sulla fetta di silicio è mantenuta in stretta vicinanza alla superficie del wafer. Assumendo di usare un fotoresist negativo, la maschera è opaca in corrispondenza delle regioni che si desidera trattare e trasparente altrove. La maschera può essere pensata come il negativo del disegno associato al particolare strato del circuito su cui stiamo lavorando. Il blocco maschera-wafer viene quindi esposto a radiazione ultravioletta e nelle regioni trasparenti della maschera il fotoresist diventa insolubile.
4. *Sviluppo del fotoresist e riscaldamento* — Il wafer viene poi sviluppato in una soluzione acida o basica per rimuovere le aree di fotoresist non esposto. Una volta che il fotoresist esposto è stato rimosso, il wafer viene riscaldato a temperature relativamente basse per rendere il fotoresist rimanente ancora più resistente.
5. *Attacco chimico* — Gli strati di materiali sottostanti il resist sono rimossi selettivamente nelle zone non coperte dal resist, mediante una serie di attacchi con soluzioni

Materiale protetto da copyright

acide, basiche o caustiche in funzione del tipo di materiale. La maggior parte del lavoro con sostanze chimiche avviene in particolari strutture dove sono preparate le soluzioni specifiche per ciascuna applicazione. A causa della pericolosità e tossicità di tali sostanze, la sicurezza e la salvaguardia dell'ambiente sono uno dei problemi principali.

6. *Rotazione, risciacquo e asciugatura* — Mediante una speciale apparecchiatura, detta SRD, *Spin, Rinse, and Dry*, il wafer viene pulito con acqua deionizzata e asciugato con azoto. La scala microscopica degli attuali dispositivi a semiconduttore è tale che anche la più piccola particella di polvere o detrito può essere fatale per l'integrità del circuito. Per prevenire ciò, i vari passi di processo devono essere eseguiti in una camera pulita dove il numero di particelle di polvere per metro cubo d'aria è compreso tra 35 e 350. Dove possibile sono impiegati sistemi robotizzati per la manipolazione dei wafer. Ciò spiega perché uno stabilimento allo stato dell'arte possa arrivare a costare diversi miliardi di dollari. Sempre per i motivi appena citati, i wafer devono essere continuamente puliti per evitare contaminazioni e rimuovere i residui dei precedenti passi di processo.
7. *Trattamento delle regioni esposte* — l'area esposta può essere soggetta a un'ampia gamma di trattamenti, quali impiantazione ionica, attacco al plasma, deposizione di metallizzazioni, ecc. Questo sarà l'argomento del prossimo paragrafo.
8. *Rimozione del fotoresist (ashing)* — Alla fine viene eseguito un attacco al plasma ad elevata temperatura per rimuovere selettivamente il fotoresist residuo senza danneggiare gli strati sottostanti.

La Figura 2-5 illustra l'incisione di uno strato di SiO_2 come esempio di processo fotolitografico. La procedura illustrata potrebbe sembrare estremamente complessa e laboriosa per la deposizione di un semplice strato di materiale, ma bisogna tenere presente che la stessa sequenza di processo viene eseguita parallelamente sull'**intera superficie del wafer**, trasferendo centinaia di milioni di tracce simultaneamente. Il parallelismo che caratterizza i processi optolitografici è ciò che ha reso fattibile e conveniente la fabbricazione di circuiti integrati molto complessi ed è proprio su di esso che si fonda il successo economico dell'industria dei semiconduttori.

La progressiva e costante miniaturizzazione dei dispositivi è un ingente onere per chi deve costruire e sviluppare le linee di produzione dei circuiti integrati a semiconduttore. Questo fardello è particolarmente significativo nel caso del processo optolitografico. Infatti, quando le dimensioni della struttura da trasferire nel wafer sono paragonabili o inferiori alla lunghezza d'onda della sorgente luminosa, è sempre più difficile ottenere la precisione e la risoluzione necessarie. Finora l'ingegneria microelettronica è stata in grado di prolungare la vita di questo processo fino all'attuale generazione dei 100 nm (0.1 μm). Ciò è stato possibile grazie a tecniche di correzione delle maschere ottiche (*OPC, optical mask correction*) che consentono di compensare i fenomeni di diffrazione presenti quando si trasferiscono strutture geometriche molto vicine o inferiori alla lunghezza d'onda della sorgente luminosa disponibile. Si tratta indubbiamente di un un costo aggiuntivo alla fabbricazione delle maschere e, nel

Materiale protetto da copyright

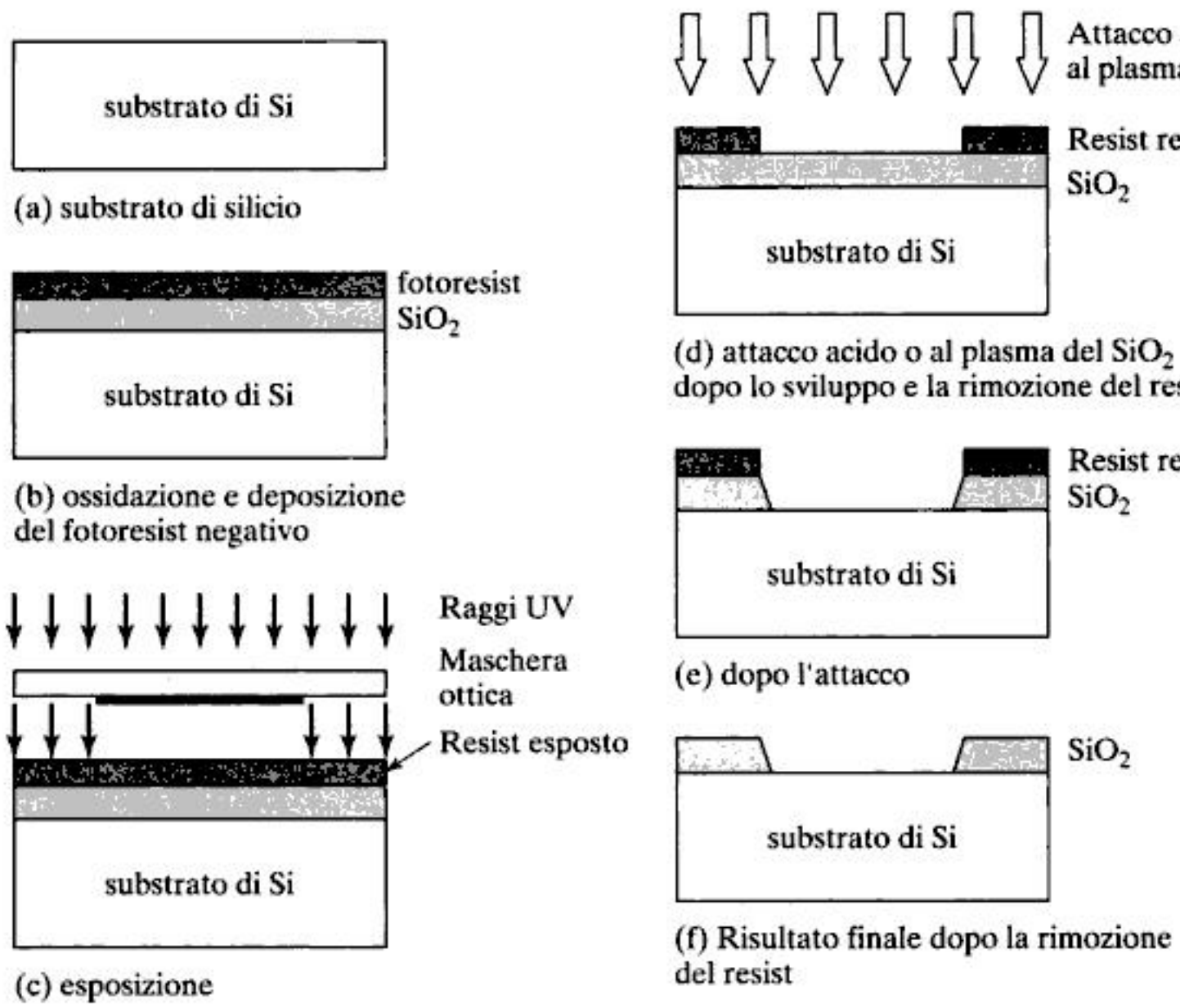


Figura 2-5 Passi di processo per l'incisione di una struttura nel SiO_2.

prossimo futuro, potrebbero essere necessarie altre tecniche in grado di garantire una maggiore risoluzione, quali la litografia EUV (*extreme ultraviolet*), la litografia a raggi X o a fascio di elettroni. Tali tecniche, sebbene già disponibili, sono attualmente meno attraenti sul piano dei costi.

2.2.3 Principali passi di processo

Diffusione e impiantazione ionica

Molti passi nella fabbricazione dei circuiti integrati richiedono una variazione del tipo o del livello del drogaggio di alcune regioni del substrato. Alcuni esempi sono la creazione delle regioni di source e di drain, i contatti di *well* e substrato, il drogaggio del polisilicio e la correzione della tensione di soglia dei transistor. Esistono due tecniche per introdurre atomi droganti nel silicio: diffusione e impiantazione ionica. In entrambe le tecniche, l'area da trattare deve essere esposta, mentre il resto della superficie deve essere protetta da uno spesso strato di materiale, tipicamente SiO_2.

Nella *diffusione ionica*, il wafer è collocato in un tubo di quarzo all'interno di una fornace riscaldata; successivamente viene introdotto nel tubo un gas contenente il materiale drogante. La temperatura molto elevata della fornace, tipicamente da 900 a 1100 °C, permette al drogante di diffondersi sia verticalmente sia orizzontalmente nelle zone in cui il silicio è

Materiale protetto da copyright

esposto al gas. La concentrazione finale del drogaggio è maggiore alla superficie e diminuisce in profondità con un profilo approssimativamente gaussiano.

Nell'*impiantazione ionica*, gli atomi di drogante sono inseriti direttamente nel reticolo del silicio sotto forma di ioni. Il sistema di impiantazione ionica tratta l'intera superficie esposta dirigendovi un fascio puro e controllato di ioni. L'accelerazione fornita agli ioni determina quanto in profondità essi penetreranno, mentre la corrente del fascio e il tempo di esposizione determinano il livello di drogaggio. Contrariamente alla diffusione, l'impiantazione ionica consente di controllare in modo indipendente la profondità e la concentrazione degli atomi droganti; peculiarità che ha consentito all'impiantazione ionica di soppiantare la tecnica di diffusione nei moderni processi manifatturieri.

Tuttavia, l'impiantazione ionica ha anche qualche inconveniente, il più importante dei quali è il danneggiamento del reticolo cristallino del substrato. Le collisioni nucleari tra ioni e atomi del reticolo durante l'impiantazione ad alta energia possono causare il dislocamento degli atomi del substrato, introducendo dei difetti strutturali nel materiale. Questo problema viene normalmente risolto sottoponendo il wafer ad un particolare trattamento, detto *annealing (recupero)*, durante il quale la fetta è riscaldata ad una temperatura di circa 1000°C per 15 - 30 minuti, e poi lasciata raffreddare lentamente. La fase di riscaldamento, aumenta le vibrazioni termiche del reticolo, permettendo ai legami spezzati o deformati di ripristinarsi.

Deposizione

Ciascun passo di un processo CMOS richiede che diversi strati di materiale siano ripetutamente depositati sulla superficie del wafer. Questi possono essere destinati a diventare parte di un dispositivo (per esempio l'isolante di gate o l'elettrodo di gate), un'interconnessione, oppure possono servire al solo scopo di proteggere gli strati sottostanti in un dato passo di processo (per esempio impiantazione ionica). Abbiamo già discusso il processo di ossidazione, con il quale viene fatto crescere uno strato di SiO_2 sulla superficie; altri materiali (nitruro di silicio, silicio policristallino, alluminio, ecc.) richiedono invece metodi diversi. Per esempio, il nitruro di silicio (Si_3N_4) è usato come strato sacrificale di protezione durante la creazione dell'ossido di campo e l'introduzione dell'impianto di isolamento (p^+-*stopper*). Normalmente il nitruro di silicio è depositato ovunque usando un processo detto CVD (*Chemical Vapor Deposition*), che è basato sulla reazione del materiale a partire dalla fase gassosa ed è catalizzato portando il sistema ad una temperatura di circa 850°C.

Al contrario, il silicio policristallino (usato come elettrodo di gate) è depositato usando un processo di deposizione chimica, nel quale il gas silano viene fatto fluire sopra la superficie del wafer ricoperta di SiO_2 in ambiente riscaldato a circa 650°C. La reazione produce un materiale amorfo o non completamente cristallino detto *polisilicio*. Per aumentare la conduttività del materiale, la deposizione deve essere seguita da un impiantazione ionica.

Le interconnessioni in alluminio sono realizzate per deposizione mediante una tecnica nota come *sputtering*: l'alluminio viene fatto evaporare nel vuoto grazie al calore generato bombardando il metallo con un fascio di elettroni o di ioni. Altri materiali per le interconnessioni come il rame richiedono altre tecniche di deposizione più complesse.

Materiale protetto da copyright

Incisione

Una volta che un materiale è stato depositato, l'incisione *(etching)* è usata selettivamente per formare strutture quali i fori per le interconnessioni e i contatti di substrato o diffusioni di source e di drain. Abbiamo già discusso il processo di incisione per via umida *(wet etching)*, che fa uso di una soluzione acida o basica. Per esempio, per incidere l'ossido di silicio si usa normalmente l'acido idrofluoridrico combinato con il fluoruro di ammonio.

Negli ultimi anni è stata introdotta con largo successo l'incisione per via secca *(dry etching)* mediante attacco al plasma *(plasma etching)*. Il wafer è polarizzato negativamente ed è posto all'interno di una camera riscaldata a circa 100°C e tenuta ad una pressione di 7.5 Pa. La camera è successivamente riempita con plasma di ioni positivi (di solito una miscela di azoto, cloro e tricloruro di bromo). In virtù della loro carica positiva le molecole del plasma, sono spinte verso il wafer verticalmente dando luogo ad un bombardamento fisico e chimico su scala microscopica che rimuove il materiale esposto. Il grande vantaggio dell'attacco al plasma rispetto all'incisione per via umida è l'ottima direzionalità che consente di realizzare strutture verticali dai contorni ben definiti.

Planarizzazione

Per depositare in maniera efficace uno strato di materiale sulla superficie del wafer è necessario che essa sia sufficientemente piana. Questo è un fattore determinante per la buona resa dei processi CMOS moderni che vedono la sovrapposizione di sempre più strati di interconnessioni metalliche. Per questo motivo, prima di depositare una interconnessione metallica su uno strato di ossido si esegue un ciclo di *planarizzazione*, detto *chemical–mechanical planarization* (CMP). In questo processo ci si avvale di un composto liquido (costituito da una soluzione in cui è sospeso un composto abrasivo come l'ossido di alluminio) in grado di "spianare" microscopicamente la superficie e ridurre l'ampiezza delle imperfezioni.

2.2.4 Descrizione esemplificata di un processo CMOS

Le fasi principali di un tipico processo CMOS sono illustrate dalla Figura 2-6. Il processo inizia con la definizione delle *regioni attive*, cioè le aree dove saranno costruiti i transistor. Tutta la superficie rimanente del wafer deve essere ricoperta da uno strato spesso di biossido di silicio (SiO_2) detto *ossido di campo (field oxide)* che funge da isolamento tra dispositivi adiacenti e può essere sia cresciuto (come nel processo indicato dalla Figura 2-1) sia depositato in apposite scanalature incise nel substrato, dette *trincee (trench)* (Figura 2-2) — da qui il nome *trench insulation (isolamento a trincea)*. Un ulteriore struttura di isolamento è realizzata aggiungendo una regione di tipo p^+, fortemente drogata, sotto l'ossido di campo, detta *channel-stop implant* (o *field implant*). Successivamente, in corrispondenza delle regioni attive sono realizzate la *p-well* e la *n-well* mediante impiantazione ionica. Per costruire un transistor NMOS, è necessario realizzare nella *p-well* (o substrato *p*) le due regioni di *source* e di *drain* di tipo *n* fortemente drogate mediante impiantazione ionica (o diffusione). Un sottile strato di SiO_2 detto *ossido di gate* separa orizzontalmente il source e il drain ed è a sua volta ricoperto da uno strato conduttivo di *polisilicio*, che forma l'elettrodo di gate del transi-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

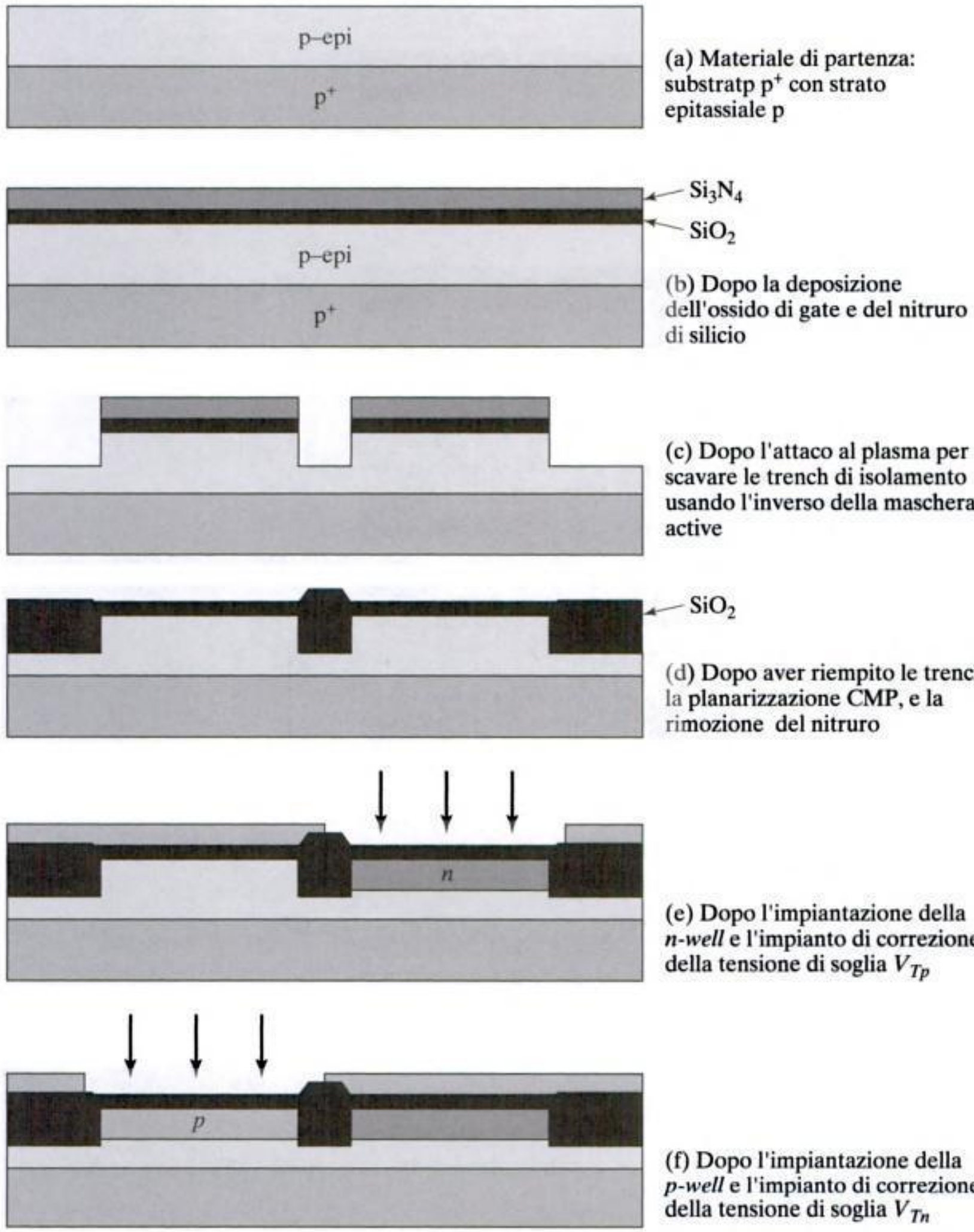
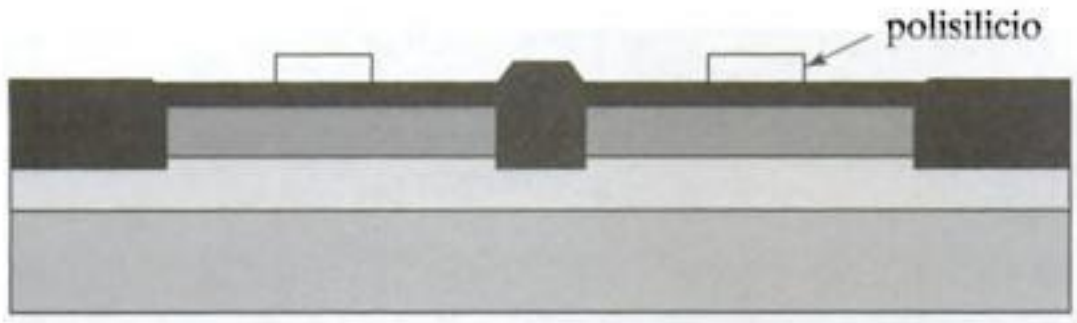


Figura 2-7 Flusso di processo per la fabbricazione di transistor NMOS e PMOS in un processo CMOS *dual-well*. Le illustrazioni sono stilizzate per una miglior comprensione e le dimensioni non sono proporzionali e conformi alla realtà.

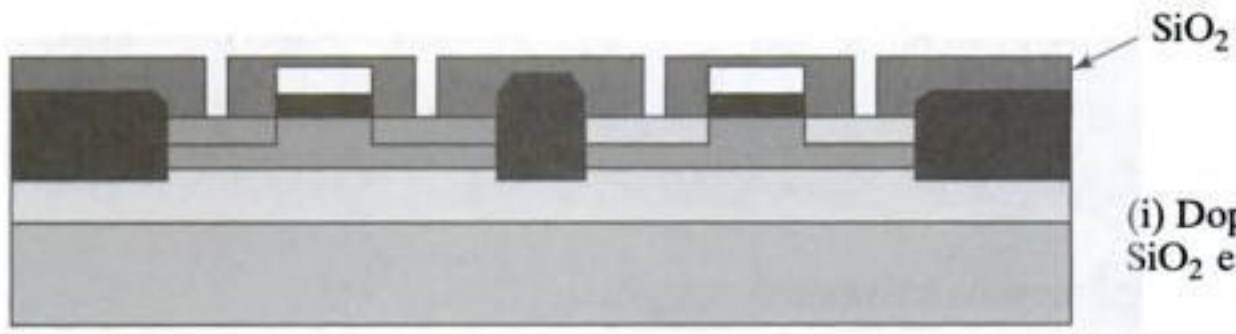
opposto) è eseguita per realizzare le *p-well*, e regolare la tensione di soglia del transistor NMOS (f). Segue la deposizione di un sottile strato di polisilicio che viene sagomato con l'ausilio della maschera appropriata; il polisilicio assolve la duplice funzione di elettrodo di gate dei transistor e di mezzo di interconnessione (g). Successivamente, mediante impiantazione ionica vengono realizzate le regioni dei source e dei drain dei transistor PMOS (p^+)

Materiale protetto da copyright

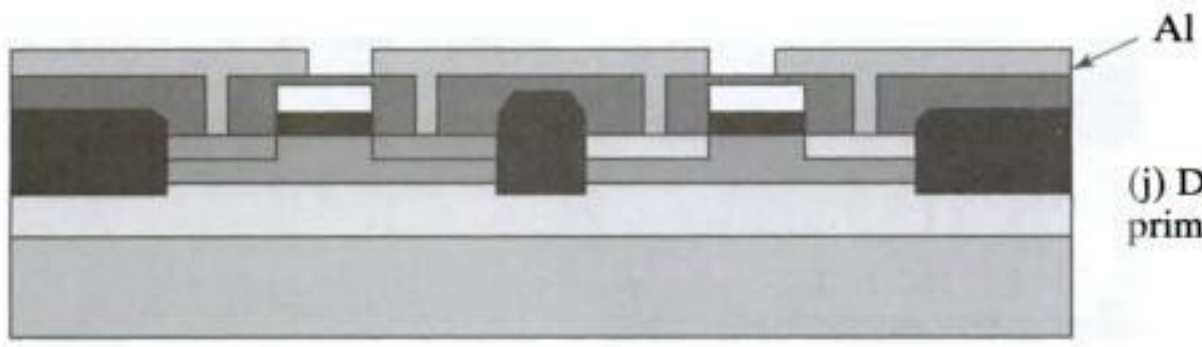


(g) Dopo la deposizione del polisilicio.

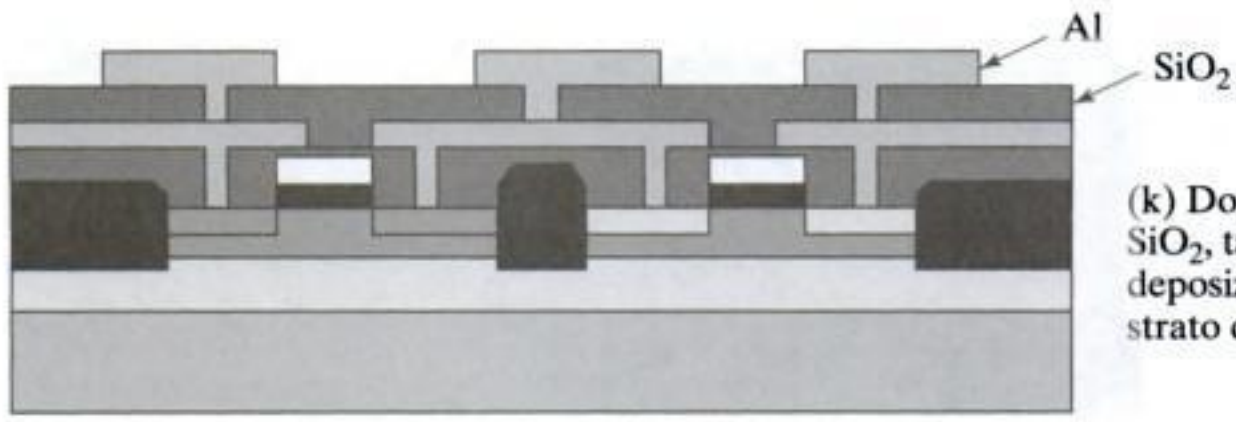
(h) Dopo l'impianazione delle regioni n^+ di source/drain e delle regioni p^+ di source/drain. Le stesse impiantazioni drogano anche il polisilicio.



(i) Dopo la deposizione del SiO_2 e il taglio dei contatti.



(j) Dopo la deposizione del primo strato di alluminio.



(k) Dopo la deposizione del SiO_2, taglio dei *via* e la deposizione del secondo strato di interconnessione.

Figure 2-7 Flusso di processo per la fabbricazione di transistor NMOS e PMOS in un processo CMOS *dual-well (continua da Figura 2-6)*

e NMOS (n^+) (h). Poi, il sottile strato di ossido di gate non coperto dal polisilicio viene rimosso.[1] Gli stessi passi di impiantazione al punto (h) vengono usati anche per drogare il polisilicio riducendone la resistività, dato che il polisilicio non drogato ha una resistività molto elevata. Si noti che di fatto è il gate di polisilicio, depositato e sagomato sopra l'area attiva, a definire l'effettiva collocazione e dimensione della regione di canale del transistor

[1]La maggior parte dei processi moderni prevede ulteriori impiantazioni per la realizzazione delle regioni LDD (lightly doped drain) e degli spacer di gate. Per semplicità abbiamo trascurato questi ulteriori passi.

Materiale protetto da copyright

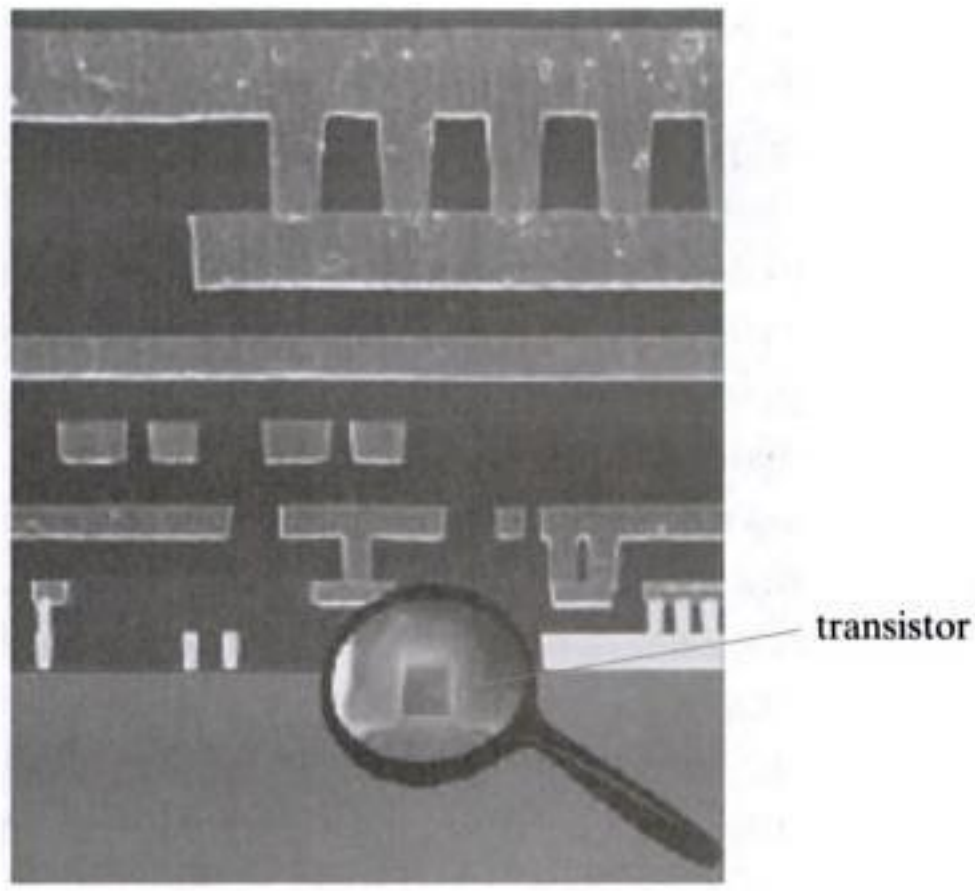


Figura 2-8 Sezione di un circuito CMOS contemporaneo.

stesso e, quindi, la posizione delle regioni di source e drain. Questa tecnica, detta processo autoallineato *(self-aligned)*, consente un'elevata precisione nel posizionare le regioni di source e drain rispetto al gate, e riduce efficacemente le capacità parassite del transistor. Il processo continua con la deposizione degli strati di interconnessioni metalliche, che consiste nella ripetizione dei seguenti passi (i–k): deposizione del materiale isolante (molto spesso SiO_2); incisione dei *contatti* o *via*, deposizione del metallo (tipicamente alluminio o rame, sebbene il tungsteno sia spesso impiegato nei primi strati); incisione e rimozione del metallo. Un passo di planarizzazione intermedio, assicura che la superficie rimanga piana, anche in presenza di numerosi strati di interconnessioni. Dopo che è stato realizzato l'ultimo strato di interconnessioni, la superficie del wafer viene protetta depositando un strato di passivazione (*overglass*). Questo strato può essere SiO_2 depositato mediante CVD, anche se spesso si usa il nitruro di silicio perché è più impermeabile all'umidità. Il passo finale del processo è l'incisione delle aperture per le piazzole di contatto *(pad)* che garantiscono la comunicazione con il mondo esterno.

La sezione trasversale del prodotto finale è illustrata dalla Figura 2-8. Si può notare che i transistor occupano solo una piccola porzione dell'estensione complessiva della struttura, la cui dimensione verticale è quasi interamente occupata dagli strati di interconnessione.

2.3 Regole di layout: il contratto tra il progettista e il costruttore

Quando le tecnologie diventano molto complesse è impensabile che il progettista possa comprendere tutte le difficoltà e i problemi dei processi di fabbricazione interpretando le relazioni tra le varie maschere. Allo scopo di facilitare e ottimizzare l'intero cammino che va dalla progettazione alla fabbricazione del circuito, viene definito un insieme di regole che permettono di semplificare la traduzione del circuito in una serie di geometrie da trasferire nel silicio e che rappresentano l'interfaccia tra il progettista e il costruttore.

Materiale protetto da copyright

Da un lato i progettisti desiderano circuiti piccoli, con elevate prestazioni e alta densità di integrazione. Dall'altro chi ha la responsabilità del processo di fabbricazione desidera una elevata resa e riproducibilità del prodotto finale. Le regole progettuali, dette anche regole di layout, sono quindi un compromesso per soddisfare entrambe le parti.

Le regole di layout forniscono delle linee guida per disegnare le varie maschere necessarie a definire le strutture geometriche da trasferire nel silicio e consistono essenzialmente in un'insieme di limiti e vincoli sulle dimensioni e le distanze reciproche tra le strutture appartenenti alla stessa maschera o a maschere diverse.

L'unità di misura fondamentale nella definizione di questo insieme di regole è detta *dimensione minima* e rappresenta la struttura più piccola che può essere correttamente trasferita dalla maschera al silicio. In genere il valore di questa dimensione minima è determinato dalla risoluzione del processo litografico, che è tipicamente basato su litografia ottica a raggi UV. Tecniche più avanzate, che impiegano la litografia a fascio di elettroni, a raggi X o EUV, offrono maggiori risoluzioni, ma a costi ben più elevati ed hanno quindi una diffusione limitata.

A parità di risoluzione litografica, le regole di layout differiscono da costruttore a costruttore e da un processo all'altro. Ciò rende molto dispendioso trasferire un progetto esistente da un processo ad un'altro. Un possibile rimedio al problema è l'impiego di tecniche di progettazione al calcolatore (CAD), che permettono la portabilità tra due processi compatibili. Un altro possibile approccio è adottare delle regole di layout *"scalabili"*. Questo secondo approccio, reso popolare da Mead and Conway [Mead80], definisce tutte le regole in funzione di un parametro globale, convenzionalmente indicato con λ. Limiti e regole sono forniti sulla base di compatibilità per un'ampia gamma di processi industriali e sono espressi in termini di relazioni funzionali di λ. Il riscalamento delle dimensioni minime da una tecnologia alla successiva è ottenuto semplicemente modificando il valore di λ: ne consegue un ridimensionamento proporzionale di tutte le dimensioni del circuito. Noti i limiti di un certo processo, a λ viene assegnato un certo valore e conseguentemente tutte le dimensioni geometriche possono essere convertite in valori assoluti. Convenzionalmente la minima dimensione realizzabile è fissata al valore 2λ: per esempio, per un processo da 0.25 µm (cioè, un processo in cui la minima dimensione permessa è 0.25 µm), λ è uguale a 0.125 µm.

Tale approccio, sebbene attraente, ha due svantaggi:

1. Il riscalamento lineare è possibile solo per variazioni limitate delle dimensioni (per esempio tra 0.25 µm e 0.18 µm). Quando l'escursione è maggiore le relazioni reciproche tra le diverse maschere cominciano a variare in modo non lineare e non possono essere adeguatamente descritte dalle stesse relazioni funzionali.
2. Le regole di riscalamento sono "conservative": esse rappresentano un compromesso valido per diverse tecnologie e quindi devono tenere conto del più restrittivo dei vincoli imposti dalle varie tecnologie. Ciò comporta spesso un sovradimensionamento delle strutture e quindi una densità di integrazione minore.

Materiale protetto da copyright

Per questi e altri motivi, le industrie non fanno uso di regole di layout scalabili.[2] Poiché la densità d'integrazione è uno degli obiettivi primari in un progetto, la maggior parte di industrie nel mondo dei semiconduttori tende ad usare regole di layout "fisse" (*micron rules*), che esprimono le relazioni tra maschere in dimensioni assolute ottimizzando al meglio le caratteristiche di un dato processo. La scalabilità e la portabilità di un progetto tra diverse tecnologie è molto più complessa e deve essere fatta o manualmente o tramite strumenti CAD avanzati.

Per questo libro abbiamo scelto come riferimento un processo CMOS da 0.25 μm. Il resto di questo paragrafo è dedicato ad una breve introduzione e ad una panoramica delle regole di layout proprie di questo processo, che rientrano nella famiglia delle regole "fisse". Un insieme completo di regole di layout comprende: un insieme di maschere o *layer* (strati); relazioni tra gli oggetti dello stesso *layer (regole intralayer)*; relazioni tra gli oggetti appartenenti a due diversi *layer (regole interlayer)*. Di seguito, discuteremo ciascuno di questi tre elementi nell'ordine.

Insieme di *layer*

Il concetto di *layer* traduce l'insieme delle maschere effettivamente usate nella realizzazione pratica del circuito, ma assolutamente intrattabili, in un semplice insieme di livelli (o strati) concettuali che costituiscono il layout del circuito facilmente visualizzabile e trattabile da parte del progettista. Dal punto di vista del progettista, tutti i circuiti CMOS sono basati sulle seguenti entità:

- Substrato e/o *well*, che sono di tipo *p* (per i dispositivi NMOS) o di tipo *n* (per i dispositivi PMOS).
- *Diffusioni* (n^+ e p^+), che definiscono le aree dove i transistor possono essere realizzati. Queste regioni sono anche dette *aree attive (active area)*. Per i contatti di substrato (o di *well*) sono necessarie delle diffusioni con drogaggio di tipo opposto. Queste regioni sono dette aree selezionate *(select region)*.
- Uno o più strati di *polisilicio (polysilicon layer)*, che sono usati per realizzare i terminali di gate dei transistor (ma servono anche per alcune interconnessioni).
- Un certo numero di livelli di *interconnessioni metalliche (metal layer)*.
- *Contatti (contact)* e *via*, che realizzano le connessioni tra livelli di metallo diversi.

Il layout consiste in un insieme di poligoni, ciascuno dei quali appartiene a un certo *layer*. La funzionalità del circuito dipende dalla combinazione di tutti questi *layer* e dalla sovrapposizione degli oggetti su *layer* diversi: per esempio, un transistor NMOS deriva dall'intersezione di un *layer* di area attiva e uno di polisilicio; una connessine tra due piste di metallo è costituita dall'intersezione di due poligoni appartenenti ai due *layer* metallici in questione e il corrispondente *layer* dei contatti. Per visualizzare queste relazioni, a ciascun *layer* è assegnato un

[2] Sebbene non molto accurate, attualmente le regole di layout scalabili sono ancora usate per valutare l'effetto dell'evoluzione tecnologica e la miniaturizzazione dei dispositivi sull'area richiesta per un dato circuito.

Materiale protetto da copyright

colore diverso (o un diverso tratteggio se rappresentati in bianco e nero). I diversi *layer* impiegati nel nostro processo CMOS di riferimento sono raffigurati nell'inserto a colori 1.

Regole *intralayer*

Un primo insieme di regole definisce le minime dimensioni e distanze fra oggetti appartenenti allo stesso layer. Per ciascun *layer* è definito un diverso insieme di dimensioni e distanze in μm. Questi limiti sono raffigurati graficamente nell'inserto a colori 2.

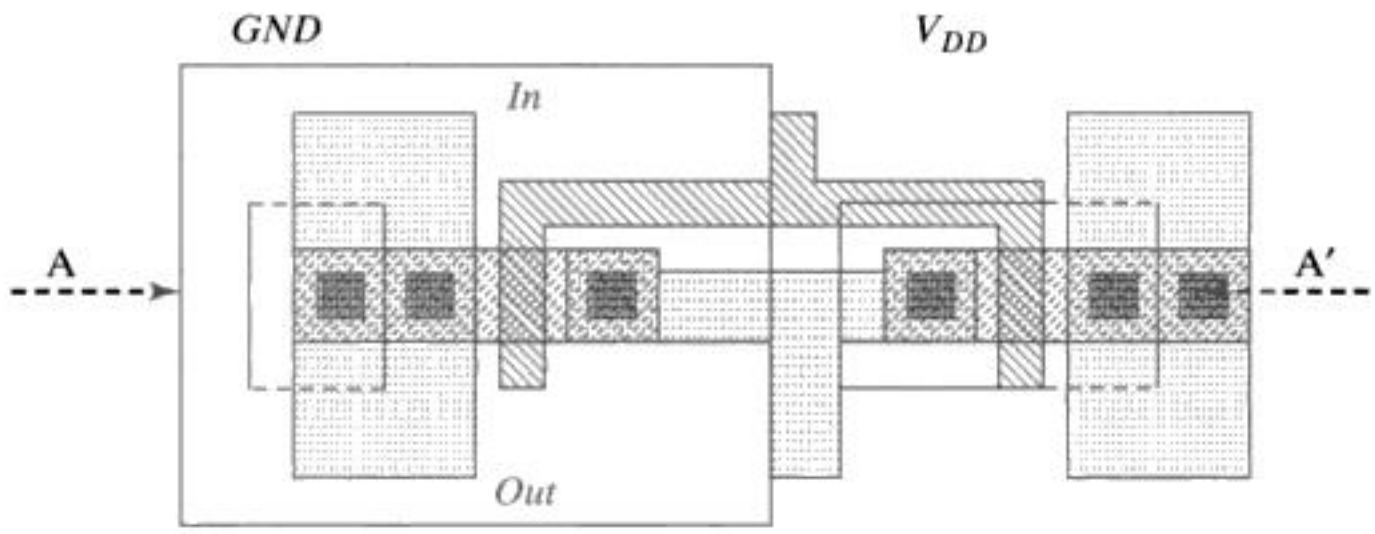
Regole *interlayer*

Le regole interlayer si riferiscono a oggetti posti su due o più *layer* e per questo tendono ad essere più complesse. Proprio perché sono coinvolti *layer* diversi, capire il layout richiede la capacità di tradurre il disegno bidimensionale in una visione tridimensionale del dispositivo. Presentiamo di seguito queste regole suddivise per gruppi:

1. *Transistor* (inserto 3). Un transistor è formato dalla sovrapposizione di una regione attiva e uno strato di polisilicio. Dalle regole intralayer risulta chiaro che la lunghezza minima del canale di un transistor è 0.24 μm (dimensione minima del polisilicio) e la larghezza minima è 0.3 μm (dimensione minima della zona attiva). In aggiunta vengono fornite altre regole riguardanti la distanza minima tra l'area attiva e il bordo della *well*, la sporgenza della regione di polisilicio oltre il bordo della zona attiva e, viceversa, la minima sporgenza della regione attiva rispetto al bordo del polisilicio.
2. *Contatti e via* (inserto 2 e 4). Un *contatto* (che forma una connessione tra una zona attiva o il polisilicio e una pista di metallo) o un *via* (che connette invece due piste di metallo in *layer* adiacenti) è formato dalla sovrapposizione dei due *layer* da collegare e il foro scavato nell'isolante che separa i due *layer*, riempito di metallo. Nel processo in esame, la minima dimensione del foro è 0.3 μm, mentre il polisilicio o la regione attiva deve estendersi di almeno 0.14 μm oltre l'area del foro stesso. Ciò impone un'area minima pari a 0.44 μm × 0.44 μm, che è maggiore dell'area minima occupata dal canale del transistor. Nel disegnare il sistema di interconnessioni è quindi consigliabile limitare al minimo indispensabile il numero di contatti tra piste metalliche su *layer* diversi. La stessa figura mostra anche la distanza minima tra i contatti così come le distanze con gli oggetti dei *layer* vicini.
3. *Contatti di well e di substrato* (inserto 5). Affinché un sistema digitale sia sufficientemente robusto e funzionale è necessario che le *well* e il substrato siano correttamente connessi alla tensione di alimentazione o alla massa. Se i contatti non sono fatti nel modo corretto vi può essere una resistenza troppo elevata tra il contatto di substrato di un transistor e la massa (o l'alimentazione), aumentando la probabilità che si verifichino fenomeni indesiderati o distruttivi, quali il latch-up. È quindi consigliabile disegnare numerosi contatti di substrato e di *well* sparsi su tutta l'area. Per formare un contatto ohmico tra la pista di massa implementata, ad esempio, in *metal1* e il substrato di tipo *p*, è necessario creare una diffusione p^+. Ciò è possibile usando la maschera *select*, che inverte il tipo di drogaggio della porzione di regione attiva selezionata. Un certo numero di regole riguardo le distanze reciproche tra i *layer* in gioco sono illustrati nell'inserto a colori 5.

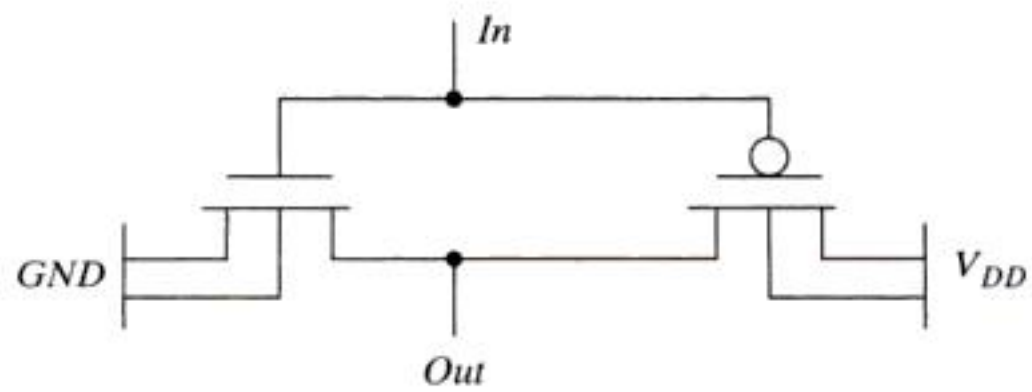
Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Layout

(b) Sezione trasversale lungo l'asse A–A′



(c) Schema circuitale

Figura 2-9 Layout di un invertitore CMOS, sezione trasversale e schema circuitale.

dei sistemi che possono essere integrati in un singolo chip si traduce nella richiesta di un numero maggiore di piedini *(pin)* di ingresso/uscita dato che il numero di connessioni verso il resto del sistema tende ad aumentare circa in modo proporzionale alla complessità circuitale del chip. Questa correlazione fu osservata per la prima volta da E. Rent di IBM (pubblicato in [Landman71]), che tradusse questa tendenza in una formula empirica, detta *legge di Rent*. Questa formula lega il numero di piedini di ingresso e uscita alla complessità del circuito, misurata come il numero di porte logiche e può essere scritta come

$$P = K \times G^{\beta} \tag{2.1}$$

dove K è il numero medio di ingressi e uscite per ciascuna porta logica, G il numero di porte logiche, β è la costante di Rent e P il numero di piedini di I/O del chip. β ha valori compresi tra 0.1 e 0.7, e dipende sopratutto dal tipo di applicazione, l'architettura e l'organizzazione del circuito come dimostrato nella Tabella 2-1: osserviamo chiaramente che i microprocessori hanno un comportamento molto diverso da quello delle memorie.

Materiale protetto da copyright

La velocità con cui aumenta il numero di piedini dei circuiti integrati varia dal 8% al 11% all'anno e si prevede che intorno al 2010 saranno richiesti circa 2000 piedini per chip. Per questo motivo i tradizionali *package dual-in-line* montati su appositi fori nella scheda, sono stati da tempo rimpiazzati da altri tipi di *package* più complessi, compatti e a montaggio superficiale o dai moduli multichip. Per il progettista è sicuramente utile conoscere le diverse soluzioni disponibili e i loro vantaggi e svantaggi.

Date le molteplici funzioni che deve svolgere, un buon *package* deve soddisfare molte caratteristiche:

- **Specifiche elettriche**. I piedini devono avere bassa capacità (sia da piedino a piedino sia da piedino a corpo del *package*), bassa resistenza e bassa induttanza. L'impedenza caratteristica deve essere ben calibrata per ottimizzare il comportamento come linea di trasmissione. A tale scopo ribadiamo che l'impedenza intrinseca del circuito integrato è molto alta.
- **Proprietà meccaniche e termiche**. La velocità di smaltimento del calore prodotto dal circuito deve essere più alta possibile. L'affidabilità e la resistenza meccanica richiede un buon compromesso tra le proprietà termiche del chip e dell'alloggiamento in cui è posto. L'affidabilità nel lungo periodo richiede anche una connessione salda e stabile tra il chip e il *package*, così come tra il *package* e la scheda.
- **Costo**. Il costo è al solito uno dei parametri da considerare con maggior attenzione nel progetto. Per esempio, sebbene i *package* ceramici abbiano prestazioni superiori rispetto ai *package* plastici, sono anche considerevolmente più costosi. Aumentare la velocità di dissipazione del calore aumenta anche il costo finale: il *package* plastico più economico è in grado di dissipare fino ad 1 W; altri *package* leggermente più costosi, ma ancora di bassa qualità, possono arrivare fino a 2 W; se le specifiche impongono prestazioni ancora maggiori è necessario ricorrere ad un *package* ceramico (più

Tabella 2-1 Costante di Rent per diversi tipi di sistemi ([Bakoglu90]).

Tipo di applicazione	β	K
Memorie statiche	0.12	6
Microprocessori	0.45	0.82
Array di porte logiche	0.5	1.9
Computer ad alta velocità (chip)	0.63	1.4
Computer ad alta velocità (scheda)	0.25	82

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

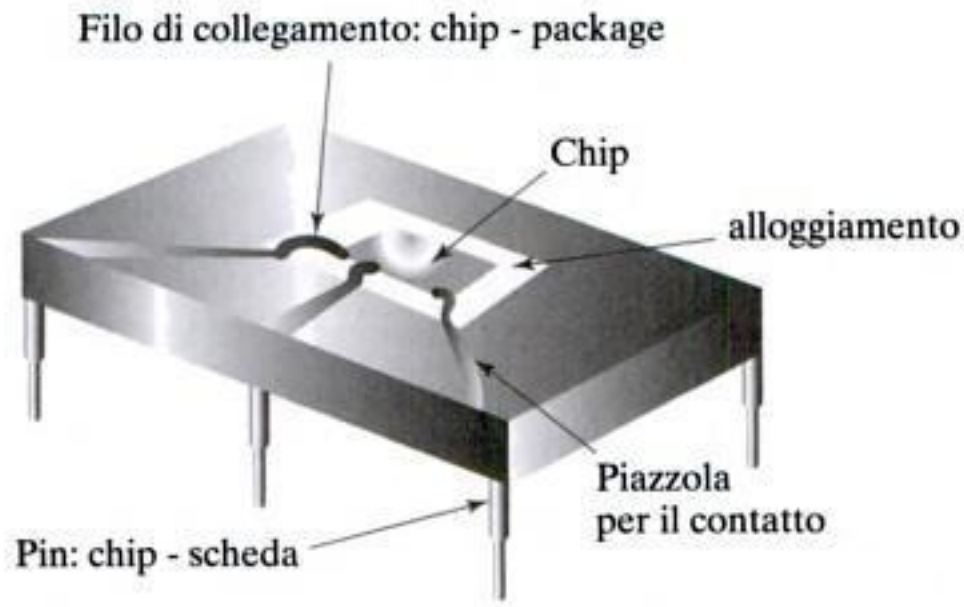


Figura 2-10 Livelli di interconnessione nel packaging dei circuiti integrati.

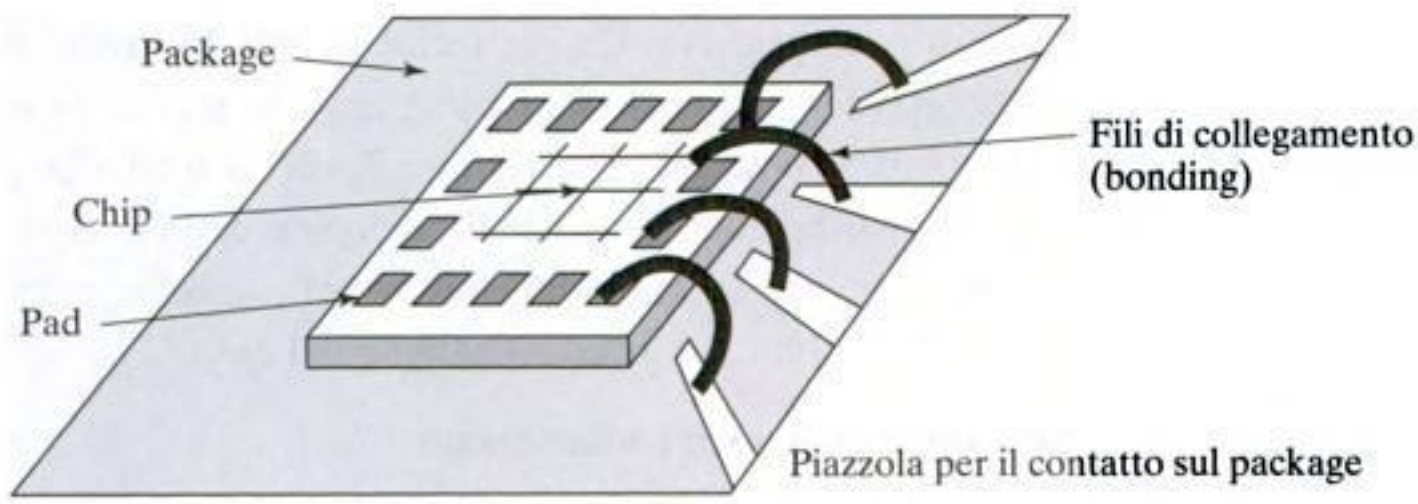


Figura 2-11 Connessione attraverso i fili di *bonding*.

mostrato come esempio in Figura 2-11. Sebbene la procedura di *bonding* abbia un alto grado di automazione, ci sono anche alcuni svantaggi:

1. I fili di collegamento devono essere saldati serialmente uno dopo l'altro con conseguente dilatamento dei tempi di fabbricazione all'aumentare del numero dei piedini.
2. Maggiore è il numero di piedini, più è oneroso trovare una configurazione di collegamenti che eviti cortocircuiti tra i fili di *bonding*.
3. A causa dell'irregolarità nei collegamenti è difficile prevedere il valore esatto degli elementi parassiti.

I fili di *bonding* non hanno proprietà elettriche eccellenti; per esempio sono caratterizzati da un elevato valore di induttanza parassita (5 nH o più) e induttanza mutua tra fili vicini. Valore tipico dell'induttanza parassita è 1 nH/mm, mentre l'induttanza dei piedini dei *package* varia da 7 a 40 nH, a seconda del tipo di *package* e della posizione del collegamento lungo il bordo del *package* stesso [Steidel83]. La Tabella 2-2 riassume i valori tipici dell'induttanza e delle capacità parassite per alcuni tipi di *package* frequentemente impiegati.

A causa di questi limiti è necessario esplorare nuove tecniche per il collegamento. In una di queste, detta *Tape Automated Bonding (TAB)*, il chip è saldato a piazzole metalliche stampate in un sottile film polimerico, generalmente poliimmide, (vedi Figura 2-12a). Le

Materiale protetto da copyright

connessioni tra i *pad* del chip e le linee di collegamento del film di supporto sono realizzate usando delle microscopiche gocce di pasta adesiva (Figura 2-12b). Il nastro polimerico può essere poi saldato al corpo del *package* in diversi modi, per esempio usando adesivi a pressione.

Il principale vantaggio della tecnica TAB è l'elevato grado di automazione: i fori nel film si prestano al trasporto dei chip mediante ruote dentate; tutte le connessioni possono essere fatte simultaneamente; la stampa delle linee metalliche di connessione sul film polimerico consente di ridurre la distanza tra le piste e quindi una maggiore densità di contatti; l'eliminazione dei lunghi fili di *bonding* migliora le proprietà elettriche del *package*. Per fare un esempio, i parametri parassiti di un componente che usa la tecnica di TAB sono: $L \approx 0.3$–0.5 nH, $C \approx 0.2$–0.3 pF, e $R \approx 50$–$200\ \Omega$ [Doane93, p. 420].

Un secondo approccio consiste nel capovolgere il chip e saldarlo direttamente al substrato del *package* usando gocce di pasta adesiva. Questa tecnica, detta *flip-chip*, è caratterizzata da ottime proprietà elettriche (vedi Figura 2-13.) Inoltre, anziché realizzare tutti i *pad* di interconnessioni al bordo del chip, è possibile posizionare i *pad* in qualsiasi parte del chip, permettendo di ottimizzare la distribuzione dei segnali di clock e delle linee di alimentazione, dato che i materiali impiegati per le interconnessioni nei *package* (rame o oro) hanno proprietà elettriche migliori dell'alluminio usato all'interno del chip.

Secondo livello di interconnessioni: dal *package* alla scheda

La tecnica tradizionalmente più usata per connettere il chip ad una scheda consiste nella saldatura del chip inserendo i piedini in fori appositamente realizzati sulla scheda (*through-hole mounting*). Una comune scheda per circuiti stampati è costruita alternando strati di rame a strati di resina isolante; su di essa vengono poi realizzati i fori ricalcando la disposizione dei piedini del chip e placcati con rame. I piedini del *package* sono infine inseriti nei fori e sal-

Tabella 2-2 Valori tipici della capacità e dell'induttanza parassite di alcuni *package* e tecniche di *bonding* (da [Steidel83], [Franzon93] e [Harper00]).

Tipo di *package*	Capacità (pF)	Induttanza (nH)
DIP plastico 68-pin	4	35
DIP ceramico 68-pin	7	20
Ball Grid Array 300 pin	1–5	2–15
Filo di bonding	0.5–1	1–2
Pasta adesiva conduttiva	0.1–0.5	0.01–0.1

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Tabella 2-3 Caratteristiche di alcuni tipi di *package*.

Tipo di *package*	Spaziatura dei contatti (valore tipico)	Numero di piedini (valore massimo)
Dual in line	2.54 mm	64
Pin grid array	2.54 mm	> 300
Small-outline IC	1.27 mm	28
Leaded chip carrier (PLCC)	1.27 mm	124
Leadless chip carrier	0.75 mm	124

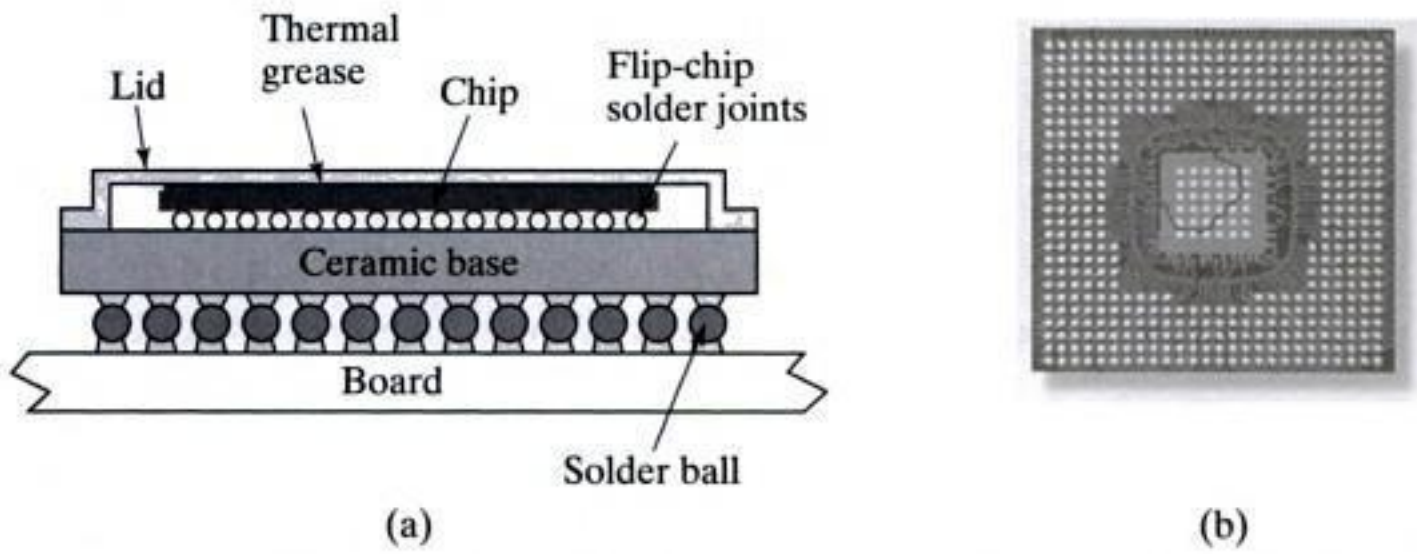


Figura 2-16 Tecnica ball grid array; (a) sezione trasversale, (b) fotografia del lato inferiore del *package*.

zione: oggigiorno, i circuiti ad alte prestazioni e alimentati a basse tensioni di alimentazione richiedono tanti piedini di massa/alimentazione quanti sono i segnali di ingresso e uscita. Quando si ha a che fare con circuiti con più di 300 terminali di ingresso e uscita è preferibile sostituire i piedini con gocce di pasta adesiva. Un esempio di questa tecnica di *packaging* è detto *ball grid array* (BGA) ed è mostrato in Figura 2-16. Sia le connessioni interne del chip al *package* sia la saldatura del *package* alla scheda sono eseguite mediante piccole gocce di materiale adesivo e conduttivo. La matrice di contatti nel lato inferiore del contenitore BGA fornisce una densità di ingressi/uscite elevata e costante indipendentemente dal numero di piedini. La spaziatura minima tra i contatti è molto bassa (fino a 0.8 mm), rendendo fattibili *package* con diverse migliaia di segnali di ingresso/uscita.

Moduli multichip: dal chip alla scheda

La complessa gerarchia di interconnessioni nel *package* dei circuiti moderni ha oramai raggiunto livelli critici in termini di numero di piedini di I/O, a causa dell'elevato grado di

Materiale protetto da copyright

integrazione e della ricerca di sempre maggiori prestazioni. Per l'immediato futuro, la tendenza è quella di ridurre il numero di livelli di interconnessioni, ad esempio il primo, cioè quello che connette il chip al *package*. Rimuovere un livello di interconnessione e saldare il chip direttamente sul piano delle interconnessioni (la scheda o un supporto ad hoc contenente più chip) offre un considerevole beneficio quando le specifiche primarie sono le prestazioni e la densità di integrazione. Questo tipo di approccio, detto *modulo multichip (MCM, multichip module)*, fornisce elevata densità di integrazione e migliori prestazioni globali del sistema.

Molte delle tecniche di *bonding* analizzate nel paragrafo precedente possono essere modificate e adattate per saldare il chip direttamente sulla scheda; tra queste il *bonding* mediante fili, la tecnica TAB e *flip-chip*. In particolare, le ultime due sono quelle preferite. Il tipo di supporto per i chip può spaziare in una vasta gamma di materiali, a seconda delle caratteristiche meccaniche, elettriche, termiche ed economiche che sono richieste. I materiali maggiormente indicati sono resine (simili a quelle usate nelle consuete schede per circuiti stampati), metalli, ceramiche e silicio. Il silicio ha il ragguardevole vantaggio di presentare caratteristiche che si adattano perfettamente alle proprietà meccaniche e termiche del chip.

Come già accennato, il principale vantaggio della tecnica MCM è l'aumento della densità e il miglioramento delle prestazioni del sistema. Un esempio in cui viene impiegato un supporto di silicio (comunemente ribattezzato *silicon on silicon*) è riportato in Figura 2-17. Il modulo illustrato, che implementa un processore per l'aviazione fabbricato dalla *Rockwell International*, contiene 53 circuiti integrati e 40 componenti discreti su un area di 5.6 × 5.6 cm con interconnessioni in alluminio. Le piste di interconnessione tra i chip del sistema

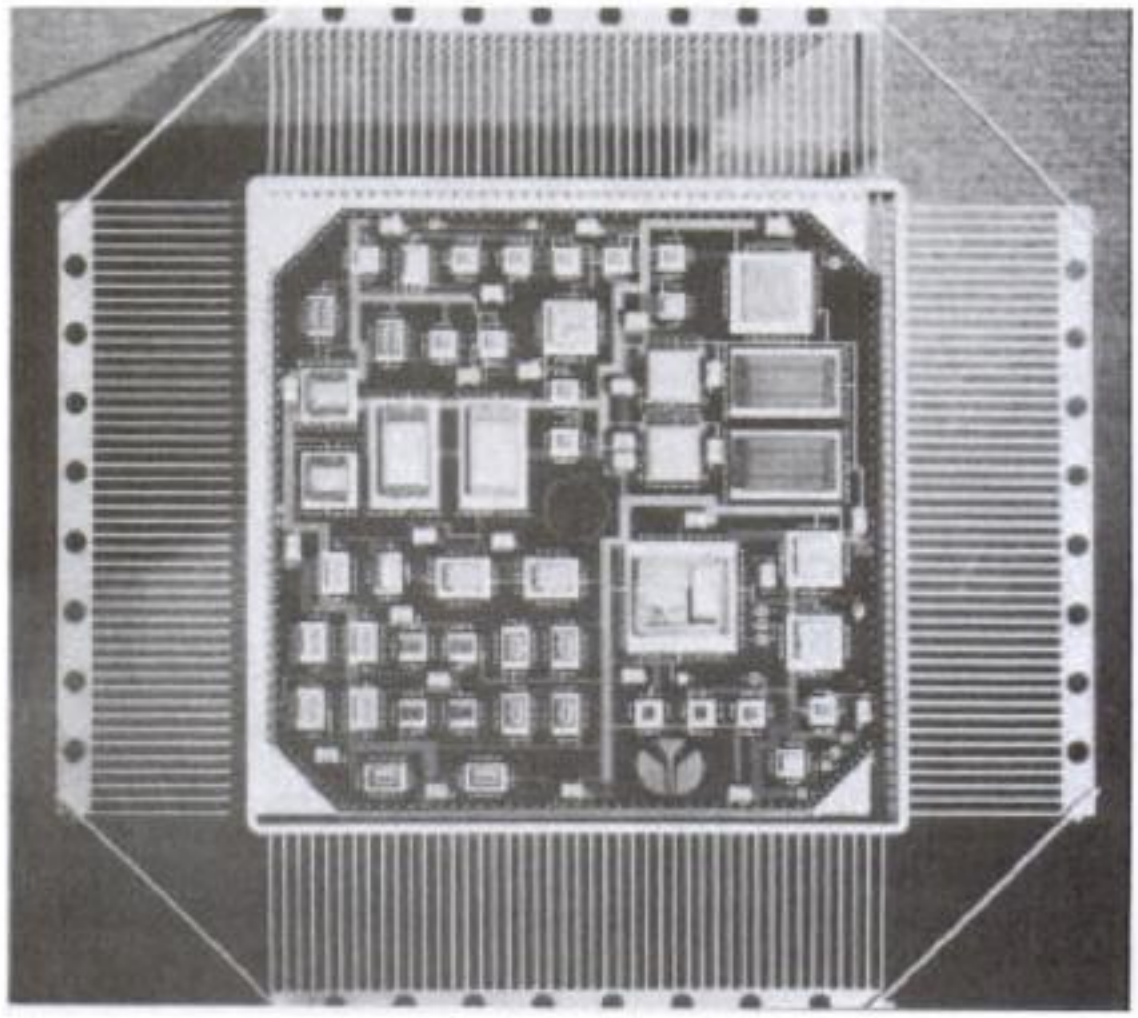


Figura 2-17 Modulo di un processore impiegato nell'aviazione. *Gentile concessione di Rockwell Collins, Inc.*

Materiale protetto da copyright

sono solo un ordine di grandezza più larghe di quelle tipicamente usate all'interno dei chip stessi, dato che la rete di connessione è realizzata e sagomata usando tecniche analoghe. Il modulo complessivo ha 180 piedini di I/O e prestazioni migliori grazie all'eliminazione dei *package* con i relativi parametri parassiti e alla ridotta lunghezza delle interconnessioni. Una goccia di pasta adesiva conduttrice, per esempio, ha valori di capacità e induttanza parassite di soli 0.1 pF e 0.01 nH, rispettivamente. La tecnologia MCM è in grado di diminuire anche il consumo di potenza, dato che possono essere eliminati o ridotti i buffer di uscita (con il relativo consumo) e non essendo più necessario pilotare gli elevati carichi capacitivi tipici dei piedini dei *package*. Allo stesso tempo, è sensibilmente ridotto anche il consumo dinamico di potenza dovuto alla commutazione delle elevate capacità di carico.

Sebbene la tecnica MCM offra notevoli benefici, lo svantaggio principale è il costo maggiore, dovuto alla necessità di qualche passo aggiuntivo di fabbricazione. Fino ad oggi il ricorso ad una tecnica così avanzata era giustificabile solo in presenza di requisiti stringenti in termini di prestazioni, densità di integrazione od occupazione di area. Recentemente i progressi e i miglioramenti tecnologici hanno reso più conveniente, anche dal punto di vista economico, l'integrazione di più chip nello stesso supporto anche per applicazioni a basso costo. Questa spinta tecnologica è ciò che sta alla base delle nuove strategie *system-in-a-package* (SIP).

2.4.3 Considerazioni termiche nel packaging

All'aumentare del consumo di potenza diventa sempre più importante dissipare in modo efficiente il calore generato all'interno del circuito integrato, dato che molti dei fenomeni di rottura di un circuito sono favoriti dalle elevate temperature. Esempi sono la corrente di perdita dei diodi polarizzati inversamente, l'elettromigrazione e la generazione di elettroni caldi (*hot-electron*). Per prevenire il verificarsi di fenomeni di rottura è necessario mantenere la temperatura entro certi limiti. Nei componenti commerciali la temperatura operativa è compresa tra 0° e 70°C, mentre nei dispositivi ad uso militare è generalmente richiesta un'escursione più ampia (da –55° a 125°C).

L'efficienza con cui viene raffredato un *package* dipende dalla resistenza termica del materiale di cui è composto, dalla sua struttura, dal tipo e dall'efficienza del sistema di raffredamento (ventole, liquido di raffredamento, ecc.). Generalmente le tecniche standard prevedono che il *package* sia lasciato in aria, impiegando al massimo delle ventole per il ricircolo dell'aria e per agevolare la dissipazione del calore. L'efficienza di smaltimento può essere migliorata ulteriormente aggiungendo delle alette di raffredamento al *package*. In applicazioni come i supercomputer o i mainframe sono adottate soluzioni più costose basate su sistemi idraulici a flusso forzato di aria, liquido o gas inerte attraverso sottili condotti a stretto contatto con la superficie del *package* per massimizzare la velocità di rimozione del calore.

Nota la resistenza termica θ del *package*, espressa in °C/W, possiamo calcolare la temperatura interna del chip usando l'*equazione di diffusione del calore*

$$\Delta T = T_{chip} - T_{amb} = \theta Q, \tag{2.2}$$

Materiale protetto da copyright

dove T_{chip} e T_{amb} sono rispettivamente la temperatura del chip e la temperatura ambiente e Q rappresenta il flusso di calore (in Watt). Notiamo come l'equazione di diffusione del calore ricordi la legge di Ohm: il flusso di calore e il gradiente termico sono rispettivamente equivalenti alla corrente e alla tensione. Costruire un modello termico del chip, del suo *package* e dell'ambiente è un lavoro complesso e per una trattazione più dettagliata rimandiamo il lettore a [Lau98 – Capitolo 3].

Esempio 2.2 Conduzione termica del *package*

Un *package* DIP a 40 piedini è caratterizzato da una resistenza termica di 38°C/W, se posto semplicemente in aria, o 25°C/W se posto in aria con ventilazione forzata. Ciò significa che un DIP può dissipare una potenza di 2 W con ventilazione naturale e 3 W con ventilazione forzata, mantendo una differenza di temperatura tra il chip e l'ambiente inferiore a 75°C. Per confronto, la resistenza termica di un *package* ceramico PGA varia da 15° a 30°C/W.

Poiché l'uso di *package* con resistenza termica sempre più bassa avrebbe costi proibitivi, mantenere limitato il consumo di potenza di un circuito diventa anche una necessità economica. Il crescente livello di integrazione e le crescenti prestazioni rendono questo compito non banale. Una relazione interessante in questo contesto è stata dedotta da Nagata [Nagata92] e fornisce un limite superiore alla complessità di integrazione e alle prestazioni, espressa dal tempo di propagazione, in funzione dei parametri termici:

$$\frac{N_G}{t_p} \leq \frac{\Delta T}{\theta E} \tag{2.3}$$

dove N_G è il numero di porte logiche nel chip, t_p il tempo di propagazione, ΔT il massimo gradiente termico tra il chip e l'ambiente, θ la resistenza termica tra essi ed E l'energia media associata alla commutazione di una porta.

Esempio 2.3 Limite termico all'integrazione

Sia $\Delta T = 100°C$, $\theta = 2.5°C/W$ e $E = 0.1$ pJ, otteniamo $N_G/t_p \leq 4 \times 10^5$ (porte/nanosecondo). In altre parole, il numero di porte logiche in un chip, quando tutte le porte lavorano simultaneamente, devono essere meno di 400 000 se la velocità di ciascuna porta è inferiore a 1 ns. Questo equivale ad un consumo di potenza pari a 40 W.

Fortunatamente, in un sistema reale non tutte le porte logiche lavorano simultaneamente e quindi il numero totale di porte logiche nel chip può essere considerevolmente più grande a seconda del numero medio di commutazioni che una porta effettivamente esegue.

Materiale protetto da copyright

Per esempio, è stato calcolato che nei PC e nei mini-computer il rapporto tra il periodo medio di commutazione e il tempo di propagazione varia da 20 a 200 [Masaki92].

Ciononostante, l'Equazione (2.3) dimostra che la dissipazione del calore e i problemi termici presentano un limite importante all'integrazione del circuito. Oggigiorno le tecniche di progettazione volte a ridurre il consumo di potenza diminuendo sia la quantità di energia dissipata sia il fattore di attività del sistema stanno guadagnando rapidamente importanza.

2.5 Le prospettive: evoluzione dei processi tecnologici

I moderni processi CMOS seguono abbastanza da vicino il flusso di operazioni descritto nei paragrafi precedenti, sebbene alcuni dei passi illustrati possano essere sostanzialmente diversi: potrebbe essere impiegata una sola *well*; potrebbe essere utilizzato l'ossido di campo invece dell'isolamento a trincea; o potrebbero essere introdotti dei passi aggiuntivi, come il *Lightly Doped Drain (LDD)*. Inoltre è prassi abbastanza comune ricoprire il polisilicio e le diffusioni di source e drain con leghe particolari come il $TiSi_2$ detta anche *siliciuro (silicide)*, allo scopo di migliorare la conduttività (vedi Figura 2-2). Questo passo è aggiunto tra i cicli *i* e *j* del nostro processo. Alcune importanti variazioni e miglioramenti tecnologici sono attualmente in corso o o sono all'orrizzonte e meritano qualche attenzione. Oltre a ciò, nei prossimi dieci anni non ci si vedono all'orizzonte grandi cambiamamenti alla tecnologia CMOS.

2.5.1 Nel breve periodo

Il rame e i dielettrici "low-k"

Un argomento abbastanza ricorrente in questo libro sarà l'impatto delle interconnessioni sulle prestazioni complessive di un sistema. Gli ingegneri di processo studiano e valutano continuamente soluzioni alternative ai tradizionali materiali (alluminio come conduttore e SiO_2 come isolante) che sono stati la base dei processi produttivi degli ultimi decenni. Nel 1998 gli ingegneri di IBM hanno sviluppato una tecnologia che ha reso possibile (anche da un punto di vista economico) la realizzazione delle interconnessioni in rame [Geppert98]. Il rame ha una resistività considerevolmente inferiore all'alluminio, ma ha anche lo svantaggio di diffondere meglio nel silicio durante il processo, degradando le caratteristiche del dispositivo. Per rimediare, è necessario proteggere il rame con materiali quali il nitruro di titanio che ne impedisca la diffusione, ma ciò richiede un processo di deposizione addizionale. Questa tecnica introdotta da IBM, detta *Dual Damascene* (Figura 2-18), prevede la realizzazione di linee di interconnessione mediante la previa creazione di canali in uno strato isolante piano, il loro riempimento con il metallo (rame in questo caso), e si conclude con un ciclo di planarizzazione. In altre parole, questa procedura è esattamente l'opposto di quella tradizionale in cui il materiale viene invece deposto in tutta la superficie e poi rimosso dalle regioni circostanti le piste.

Oltre all'impiego di materiali con resistività inferiore per le interconnessioni, a partire dalla generazione tecnologica CMOS da 0.18 μm sono stati introdotti anche isolanti con

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

principali sono la riduzione delle capacità parassite e le migliori caratteristiche come interruttore del transistor. Per esempio, è stato dimostrato dai ricercatori IBM che la realizzazione dello stesso sistema in tecnologia CMOS SOI (lasciando invariati tutti gli altri parametri, come lo spessore dell'ossido di gate e la lunghezza di canale) fornisce un miglioramento del 22% nelle prestazioni rispetto allo stesso circuito realizzato con un processo CMOS convenzionale [Allen99]. Il maggior ostacolo alla diffusione su larga scala della tecnologia SOI è stato per molto tempo il costo necessario per la preparazione di un substratro di alta qualità, ma questo scenario è mutato alla fine degli anni '90 aprendo la strada alla diffusione della tecnologia SOI.

2.5.2 Nel lungo periodo

Estendere la vita della tecnologia CMOS oltre i prossimi dieci anni, andando ben oltre la generazione dei 100 nm, richiede una revisione delle tecnologie di processo e della struttura stessa del dispositivo. Attualmente siamo già testimoni della nascia di nuovi dispositivi (come i transistor organici, gli interruttori molecolari, e i dispositivi quantistici). Sebbene non siamo ancora in grado di stabilire quali di questi sarà l'approcio vincente in futuro, vale la pena di menzionare gli sviluppi più interessanti.

Circuiti intregrati tridimensionali

Una delle sfide più avvincenti che nasceranno a seguito della cresciente densità di integrazione è quella delle connesssione tra i dispositivi e blocchi di un circuito, il cui numero sta crescendo esponenzialmente. Un modo per affrontare il problema è quello di introdurre ulteriori strati di dispositivi e impilarli uno sopra l'altro alternati a strati di interconnessioni, come illustrato in Figura 2-20. Ciò permetterebbe di collocare memorie ad alta densità sopra i blocchi logici di un microprocessore, riducendo la distanza tra i moduli computazionali e quelli di immagazinamento delle informazioni, e quindi i tempi di ritardo [Souri00]. Inoltre, dispositivi con diversi tipi di substrato, tensioni di alimentazione o prestazioni, possono essere posti in strati diversi. Il livello attivo più alto può, per esempio, essere riservato alla realizzazione di ricevitori ottici, o dispositivi MEMS *(Micro Electro-Mechanical Systems)*, che fungono da sensori o interfaccia per segnali RF.

Sebbene questo approcio possa sembrare promettente, vi sono una serie di problemi che devono trovare una soluzione affinché possa essere attuato approcio attuabile. La dissipazione del calore è una delle questioni più stringenti; un'altra sfida è quella di ottenere rese accettabili. Su queste tematiche si stanno facendo molti progressi e l'integrazione tridimensionale potrebbe essere alle porte, ma prima di arrivare ad una concreta realizzazione si dovrà passare per soluzioni intermedie. Una delle alternative, detta integrazione 2.5D, è di unire due wafer finiti uno sull'altro in modo tale che i circuiti realizzati sulla superficie si sovrappongano perfettamente. Per connettere elettricamente i due chip è necessario incidere dei contatti dopo la fase di metallizzazione. Il vantaggio di questa tecnologia risiede nelle proprietà elettriche molto simili dei dispositivi su entrambi gli strati e sul fatto che i processi di fabbricazione sono indipendenti l'uno dall'altro dato che i wafer sono realizzati separata-

Materiale protetto da copyright

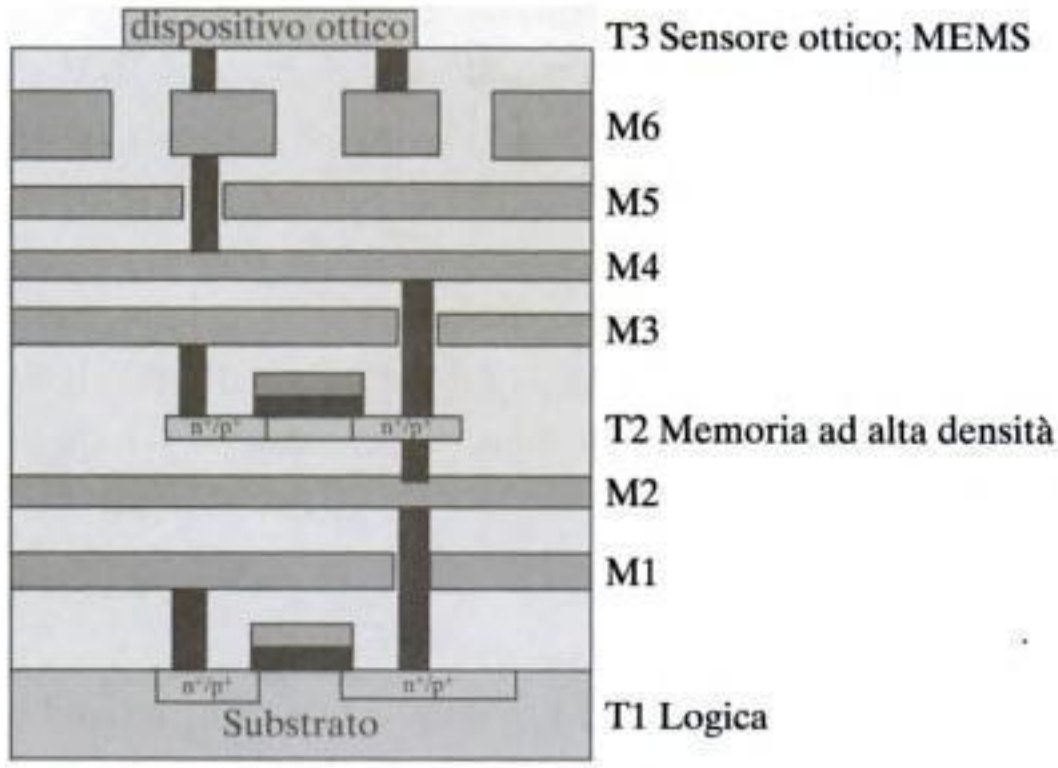


Figura 2-20 Esempio di integrazione tridimensionale. Diversi strati attivi (T*), che implementano memorie e stadi di I/O, sono inseriti tra gli strati di interconnessioni metalliche (M*).

mente e successivamente uniti. Il maggior difetto di questa tecnica è la scarsa precisione nell'allineamento (± 2 μm), che limita il numero di interconnessioni realizzabili tra un chip e l'altro.

Lo scenario che emerge da queste visioni futuristiche è che la linea che unisce chip, substrato, *package* e scheda è confusa e nebulosa. I progettisti di questi sistemi su singolo chip o su singolo *package* dovranno considerare simultaneamente tutti questi aspetti.

2.6 Riepilogo

Questo capitolo ha presentato una visione globale dei processi manifatturieri e del packaging dei circuiti integrati CMOS:

- I processi di fabbricazione dei circuiti integrati richiedono molti passi, ciascuno dei quali consiste in una serie di operazioni fondamentali. Alcuni di questi passi e/o operazioni, come l'esposizione e lo sviluppo nella fotolitografia, la deposizione e l'incisione dei materiali, sono ripetutamente eseguiti nel corso dell'intero processo.
- Le *maschere ottiche* formano l'interfaccia tra gli aspetti tecnologici peculiari dei processi di fabbricazione e il circuito che il progettista vuole vedere trasferito nel silicio.
- Le *regole di layout* definiscono i vincoli in termini di dimensioni e spaziatura minime che il progettista del circuito deve rispettare affinché il sistema sia correttamente funzionante. Queste regole rappresentano l'interfaccia tra i progettisti e gli ingegneri di processo.
- Il *package* è l'interfaccia tra il circuito implementato nel chip di silicio e il mondo esterno e ha un impatto determinante sulle prestazioni, sull'affidabilità, sulla robustezza e sul costo di un circuito integrato.

Materiale protetto da copyright

2.7 Approfondimenti

Negli ultimi decenni sono stati pubblicati molti libri sui processi di fabbricazione dei semiconduttori. Un'eccellente panoramica sullo stato dell'arte dei processi CMOS è: "J. Plummer, M. Deal e P. Griffin *Silicon VLSI Technology*" [Plummer00], mentre una presentazione grafica visuale dei diversi passi di processo può esssere trovata su web in [Fullman99]. Altre fonti di informazione sono le *Transactions on Electron Devices (IEEE)*, e gli atti del convegno *IEDM (International Electron Device Meeting)*. Vari compendi di informazioni aggiornate e approfondimenti sul *packaging* possono essere trovati in [Doane93], [Harper00] e [Lau98].

Riferimenti scientifici

[Allen99] D. Allen, et al., "A 0.2 μm 1.8 V SOI 550 MHz PowerPC Microprocessor with Copper Interconnects," *Proceedings IEEE ISSCC Conference*, vol. XLII, pp. 438–439, February 1999.

[Bakoglu90] H. Bakoglu, *Circuits, Interconnections and Packaging for VLSI*, Addison-Wesley, 1990.

[Doane93] D. Doane, ed., *Multichip Module Technologies and Alternatives*, Van Nostrand-Reinhold, 1993.

[Eaglesham 99] D. Englesham, "0.18 μm CMOS and Beyond," *Proceedings 1999 Design Automation Conference*, pp. 703–708, June 1999.

[Franzon93] P. Franzon, "Electrical Design of Digital Multichip Modules," in [Doane93], pp 525–568, 1993.

[Fullman99] Fullman Kinetics, "The Semiconductor Manufacturing Process," *http://www.fullman-kinetics.com/semiconductors/semiconductors.html*, 1999.

[Geppert98] L. Geppert, "Technology—1998 Analysis and Forecast," *IEEE Spectrum*, vol. 35, no. 1, p. 23, January 1998.

[Harper00] C. Harper, Ed., *Electronic Packaging and Interconnection Handbook*, McGraw-Hill, 2000.

[Landman71] B. Landman and R. Russo, "On a Pin versus Block Relationship for Partitions of Logic Graphs," *IEEE Trans. on Computers*, vol. C-20, pp. 1469–1479, December 1971.

[Lau98] J. Lau et al., *Electronic Packaging—Design, Materials, Process, and Reliability*, McGraw-Hill, 1998.

[Masaki92] A. Masaki, "Deep-Submicron CMOS Warms Up to High-Speed Logic," *Circuits and Devices Magazine*, November 1992.

[Mead80] C. Mead and L. Conway, *Introduction to VLSI Systems*, Addison-Wesley, 1980.

[Nagata92] M. Nagata, "Limitations, Innovations, and Challenges of Circuits and Devices into a Half Micrometer and Beyond," *IEEE Journal of Solid State Circuits*, vol. 27, no. 4, pp. 465–472, April 1992.

[Plummer00] J. Plummer, M. Deal, and P. Griffin, *Silicon VLSI Technology*, Prentice Hall, 2000.

[Steidel83] C. Steidel, *Assembly Techniques and Packaging*, in [Sze83], pp. 551–598, 1983.

[Souri00] S. J. Souri, K. Banerjee, A. Mehrotra and K. C. Saraswat, "*Multiple Si Layer ICs: Motivation, Performance Analysis, and Design Implications*," Proceedings 37th Design Automation Conference, pp. 213–220, June 2000.

Materiale protetto da copyright

Materiale protetto da copyright

Layout dei circuiti digitali

Come disegnare un layout corretto

Verifica del layout

La crescente complessità dei circuiti integrati ha reso indispensabile il ricorso a strumenti di progettazione automatica, e sta conducendo il progettista verso livelli di astrazione sempre più elevati. Tuttavia, quando le prestazioni o la densità di integrazione sono l'obiettivo primario, il progettista non ha altra scelta che ricavare il layout e ottimizzare la topologia circuitale manualmente. Questo approccio, detto *progettazione dedicata (custom)*, è estremamente dispendioso in termini di tempo e risorse e si traduce nella dilatazione dei tempi richiesti l'immissione sul mercato. Di conseguenza, esso trova una giustificazione economica solo in determinate condizioni:

- Tali blocchi dedicati possono essere riutilizzati molte volte, come, per esempio, una libreria di celle.
- I costi possono essere ammortizzati su un gran numero di pezzi venduti, per esempio in caso di applicazioni come i microprocessori e le memorie a semiconduttore.
- Il costo non è il criterio di progettazione primario,[1] come nel caso di applicazioni per lo spazio o strumentazione scientifica.

[1] Queste applicazioni stanno diventando sempre più rare.

Materiale protetto da copyright

Con l'evoluzione degli strumenti di progettazione automatica, la progettazione dedicata ha visto la sua fetta di mercato ridursi di anno in anno. Anche nei microprocessori ad alte prestazioni, una porzione considerevole dei circuiti è disegnata utilizzando un approccio *semi-automatico* (*semi-custom*). Solo i moduli più critici a cui sono richieste le prestazioni più elevate (come le unità logico-aritmetiche intere o a virgola mobile) sono realizzate e ottimizzate manualmente.

Sebbene il grado di automazione nella progettazione dedicata sia minimo, alcuni strumenti di progettazione assistita si sono mostrati ugualmente indispensabili. Insieme ai simulatori circuitali, questi programmi formano il cuore di qualsiasi ambiente di progettazione di circuiti integrati, e sono i primi strumenti che incontrerà chi aspiri a diventare un progettista di circuiti.

Editor di layout

L'editor di layout è il primo degli strumenti di lavoro di un progettista e il suo scopo principale è la generazione della rappresentazione fisica del circuito, data una certa topologia. Di fatto, tutti i pacchetti di progettazione automatica includono uno strumento di questo tipo. Il più noto tra questi è stato MAGIC sviluppato all'Università della California a Berkeley [Ousterhout84], che è ampiamente distribuito. Anche se MAGIC non ha resistito all'evoluzione delle tecnologie software, lo hanno fatto alcuni dei suoi discendenti. In questo libro useremo un editor di layout, chiamato **max**, un discendente di MAGIC sviluppato da MicroMagic [mmi00]. Una schermata tipica di **max** è mostrata in Figura A-1 e illustra la funzione basilare dell'editor di layout: disegnare poligoni su diversi livelli di maschere in modo tale da ottenere la rappresentazione fisica di un circuito funzionante.

Layout simbolico

Poiché disegnare il layout fisico di un circuito occupa una parte considerevole del tempo dedicato al progetto di una nuova cella o di un nuovo componente, sono continuamente richieste tecniche in grado di accelerare questa fase. Tra queste, il *layout simbolico* ha riscosso molta popolarità nel corso degli anni. Questo tipo di approccio consiste in una sorta di stenografia del layout: per una data struttura da disegnare, il progettista usa una notazione semplificata che indica solo la posizione *relativa* delle varie parti che la compongono (transistor, contatti, piste metalliche). Le coordinate *assolute* sono invece determinate automaticamente dal programma utilizzando un *compilatore* che traduce le regole di layout in una serie di vincoli sulla posizione della struttura ed esegue un'ottimizzazione rendendo minima l'area occupata e/o altri parametri progettuali [Hsueh79, Weste93].

Un esempio di notazione simbolica per la topologia circuitale è detta *diagramma stick* ed è riportata in Figura A-2. Le diverse entità che compongono il layout sono adimensionali, essendo rilevante solo la posizione reciproca. Il vantaggio di questa tecnica è che il progettista non deve preoccuparsi delle regole di layout nè delle dimensioni dei vari poligoni, perché è compito del compilatore garantire la correttezza del layout fisico finale. Un altro vantaggio dell'approccio simbolico è che le celle possono essere adattate all'ambiente circostante, per esempio, impostando automaticamente la spaziatura tra le celle in base alla loro dimensione.

Materiale protetto da copyright

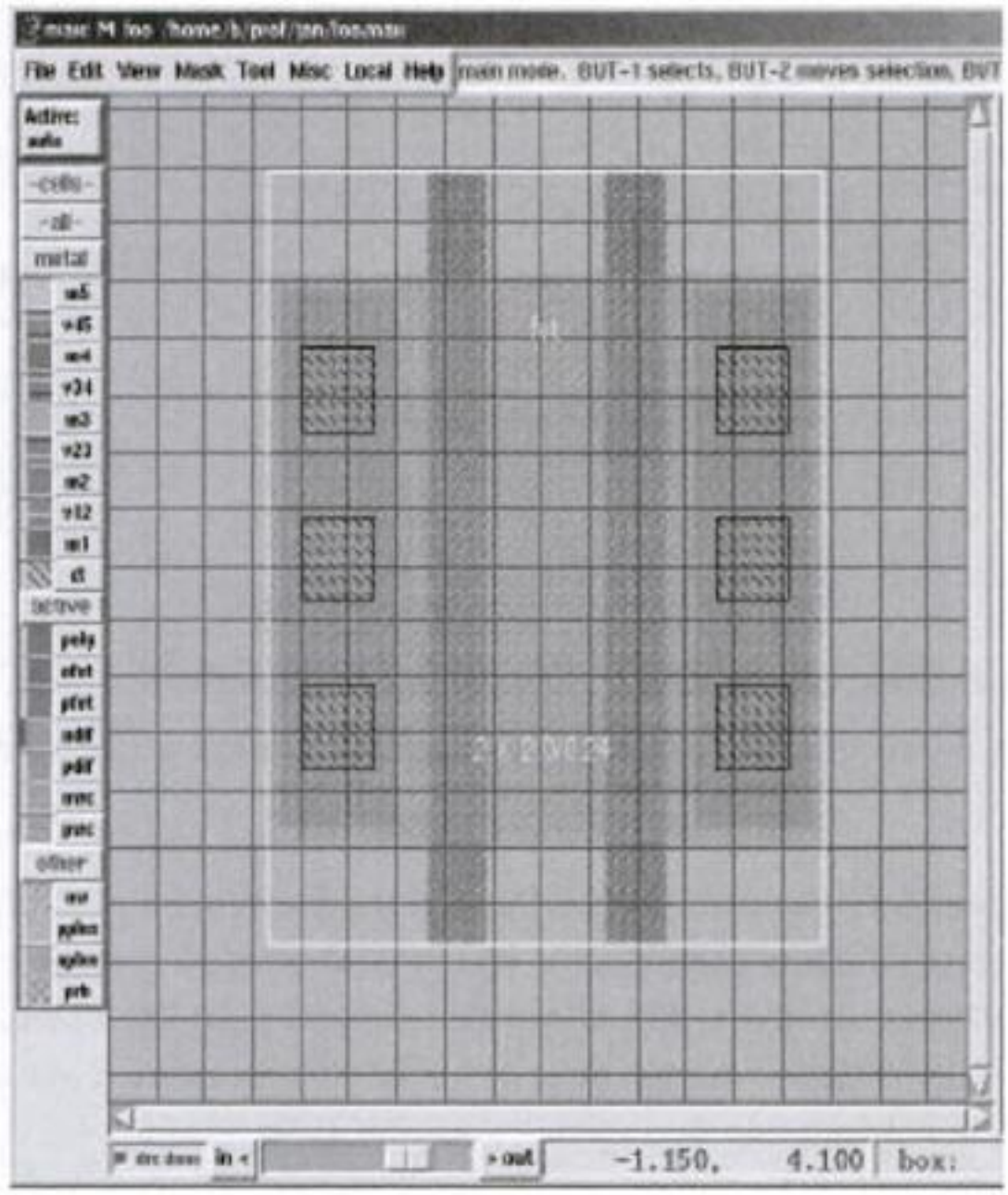


Figura A-1 Una tipica schermata di **max**. In questo esempio vediamo come sono disegnati due transistor NMOS in serie. Il menu a sinistra permette di selezionare un particolare layer e posizionare un poligono.

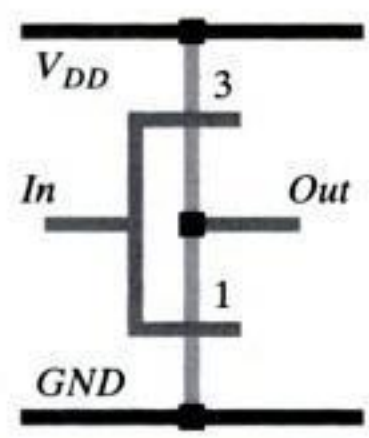


Figura A-2 Diagramma simbolico di un invertitore CMOS. I numeri rappresentano i fattori di forma (*W*/*L*) dei transistor.

Consideriamo l'esempio di Figura A-3 (tratto da [Croes88]), dove le celle originali hanno altezza diversa e la posizione dei terminali non riproduce uno schema periodico e coerente. Connettere siffatte celle richiede piste aggiuntive. Impiegando un layout simbolico le celle potrebbero essere ottimizzate facendo combaciare l'una con l'altra, senza alcuno sforzo aggiuntivo da parte del progettista.

Lo svantaggio dell'approccio simbolico è che il risultato finale della compilazione è spesso imprevedibile e meno compatto di quello che sarebbe stato ottenuto usando un

Materiale protetto da copyright

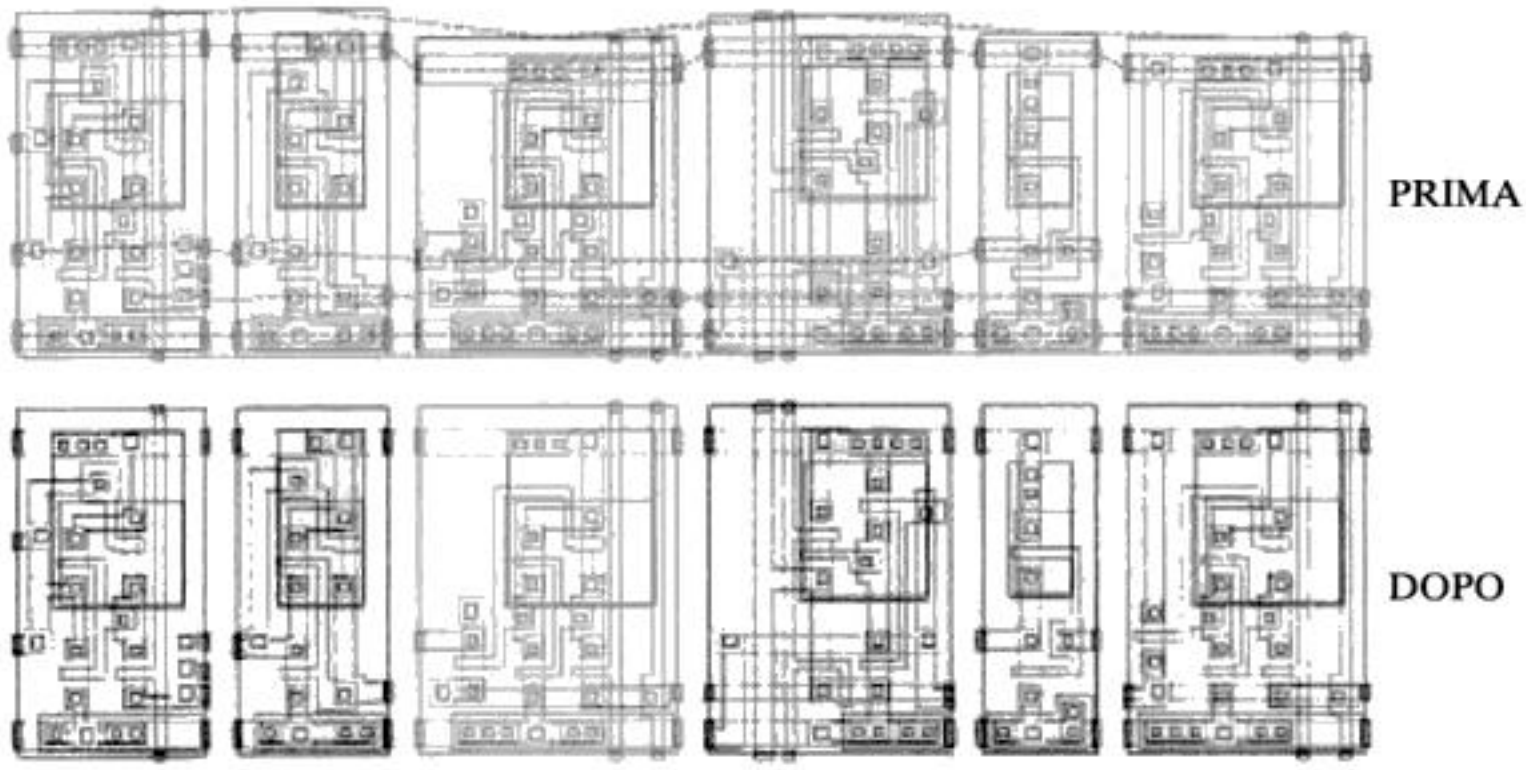


Figura A-3 Ottimizzazione automatica della spaziatura tra celle (*pitch matching*) grazie al layout simbolico.

approccio completamente manuale. Ciò ha impedito a questa tecnica di diventare il principale strumento di progettazione, anche se il layout simbolico rimane un'ottima tecnica per tracciare la prima bozza di layout di celle e componenti completamente nuovi. Inoltre il layout simbolico è il sostegno basilare delle tecniche di generazione automatica del layout che saranno descritte nel Capitolo 8.

Verifica delle regole di layout

Nel Capitolo 2 sono state introdotte le regole di layout come una serie di vincoli che, se rispettati, garantiscono che il circuito fabbricato sarà corretto, cioè conforme alle specifiche e alla funzionalità richieste (senza cortocircuiti, collegamenti mancanti, ecc.) Quindi il requisito primario del layout fisico è che esso sia conforme a tali regole. La conformità deve essere verificata mediante uno strumento apposito detto *DRC* (*design-rule checker*), che riceve in ingresso il layout fisico di un circuito e un file, fornito dal costruttore, con la descrizione delle regole di layout e contenente tutte le informazioni necessarie della tecnologia. Dato che il layout di un circuito complesso può contenere milioni e milioni di poligoni, ciascuno dei quali deve essere verificato e messo in relazione con gli altri, la proprietà fondamentale che caratterizza un DRC è sicuramente l'efficienza di elaborazione (la verifica di un chip complesso può richiedere ore o giorni). Un modo per ottimizzare l'intero procedimento è quello di preservare la struttura gerarchica del sistema anche a livello di layout fisico. Ad esempio, se una cella è usata molteplici volte in un circuito, può essere controllata una sola volta. Oltre ad accelerare la fase di verifica, l'uso di una struttura gerarchica può anche fornire maggiori informazioni nei messaggi di errore, dato che il layout fisico rispecchia la stessa struttura del circuito.

Gli strumenti di DRC possono operare in due modi: (1) *on-line,* analizzando il layout in tempo reale durante la fase stessa di stesura della cella e segnalando subito le violazioni alle regole. Per esempio, **max** ha uno strumento di verifica integrato. Un esempio di DRC on-line

Materiale protetto da copyright

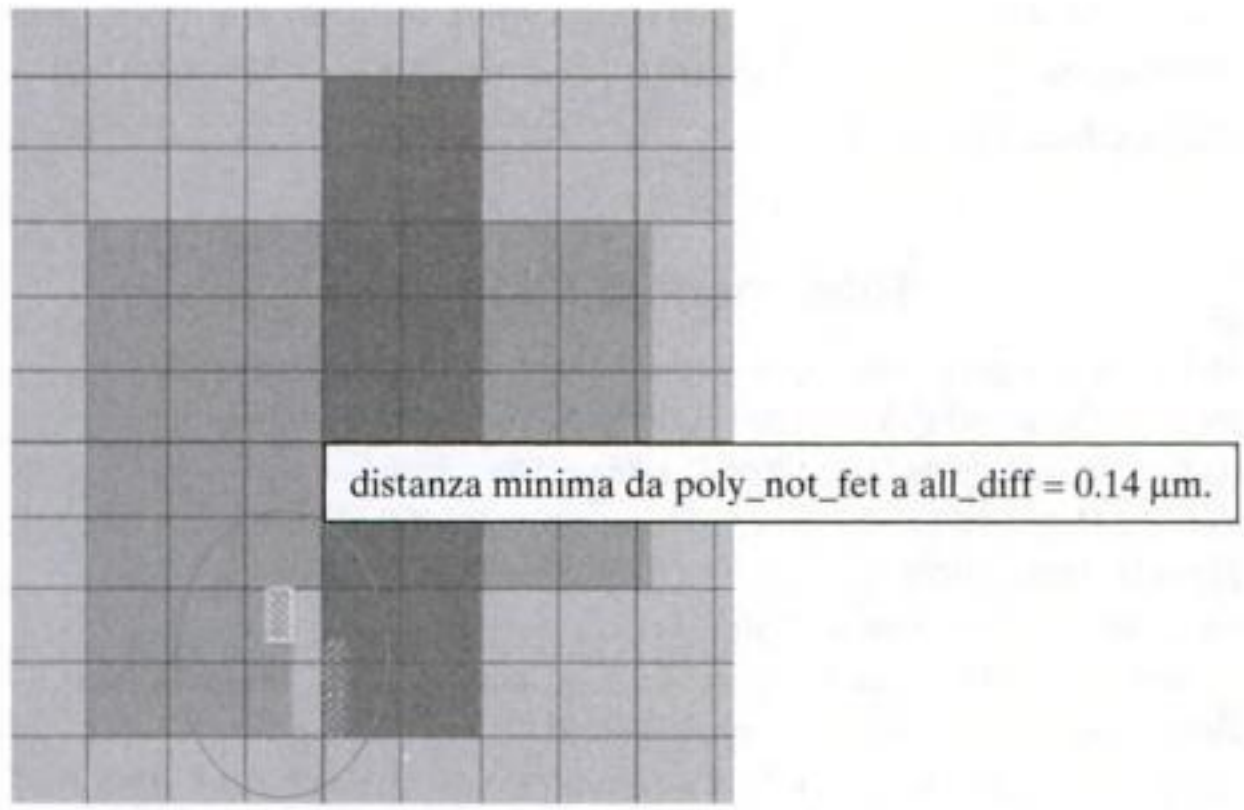


Figura A-4 Verifica delle regole di layout in tempo reale. I punti bianchi segnalano la presenza di un errore. Le informazioni su quale regola è stata violata possono essere ottenute con un semplice click del mouse.

è mostrato in Figura A-4. (2) Analizzando il layout dell'intero chip, dopo che il layout fisico è stato disegnato, ossia prima che la descrizione delle maschere sia mandata al costruttore.

Estrazione del circuito

Un altro strumento fondamentale di un pacchetto CAD è un programma di estrazione del circuito, ossia in grado di ricavare lo schema circuitale a partire dal layout fisico. Analizzando tutti i layer e le loro intersezioni lo strumento di estrazione ricostruisce la rete di transistor e le dimensioni di tutti i dispositivi e interconnessioni. Lo schema circuitale così ottenuto può essere usato per verificare che il layout disegnato implementi effettivamente la funzione desiderata. Inoltre il circuito estratto contiene anche tutte le informazioni sugli elementi parassiti (capacità e resistenze delle piste o delle diffusioni) permettendo una simulazione e un'analisi più accurata. La complessità dell'estrazione dipende in larga parte dal tipo di informazioni richieste. Generalmente sono estratte la rete di transistor e le capacità delle interconnessioni riferite alla massa; l'estrazione delle resistenze parassite delle piste necessita invece di una maggior capacità computazionale. Il problema delle interconnessioni e delle estrazioni dei parassiti è di cruciale importanza soprattutto per i circuiti ad alta velocità dove non basta stimare le capacità parassite delle piste, ma è richiesta anche la resistenza e, per le applicazioni più spinte, l'induttanza parassita. Sfortunatamente, per quest'ultima è richiesta un'analisi tridimensionale del circuito ed è fattibile solo per circuiti (o parti di circuito) di modeste dimensioni. Comunque, lo sviluppo di algoritmi ottimizzati ha contribuito alla riduzione dei tempi di estrazione del circuito e dei parassiti.

A.1 Approfondimenti

Maggiori informazioni sugli editor di layout MAGIC e **max** possono essere trovate sul sito web di questo libro. Sono stati pubblicati molti libri che sviluppano un'analisi approfondita

Materiale protetto da copyright

sul layout e la verifica di correttezza, e possono essere di grande aiuto ai progettisti principianti. Per menzionarne qualcuno, [Clein00], [Uyemura95], e [Wolf94] offrono una trattazione comprensiva e ben illustrata.

Riferimenti bibliografici

[Clein00] D. Clein, *CMOS IC Layout—Concepts, Methodologies, and Tools*, Newnes, 2000.

[Croes88] K. Croes, H. De Man, and P. Six, "CAMELEON: A Process-Tolerant Symbolic Layout System," *Journal of Solid State Circuits*, vol. 23 no. 3, pp. 705–713, June 1988.

[Hsueh79] M. Hsueh and D. Pederson, "Computer-Aided Layout of LSI Building Blocks," *Proceedings ISCAS Conf.*, pp. 474–477, Tokyo, 1979.

[mmi00] MicroMagic, Inc, *http://www.micromagic.com*.

[Ousterhout84] J. Ousterhout, G. Hamachi, R. Mayo, W. Scott, and G. Taylor, "Magic: A VLSI Layout System," *Proc. 21st Design Automation Conference*, pp. 152–159, 1984.

[Uyemura95] J. Uyemura, *Physical Design of CMOS Integrated Circuits Using L-EDIT*, PWS Publishing Company, 1995.

[Weste93] N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design—A Systems Perspective*, Addison-Wesley, 1993.

[Wolf94] W. Wolf, *Modern VLSI Design—A Systems Approach*, Prentice Hall, 1994.

Materiale protetto da copyright

CAPITOLO

I dispositivi

Analisi qualitativa dei dispositivi MOS

Modelli semplificati dei componenti per l'analisi manuale

Modelli dettagliati dei componenti per SPICE

Conseguenze delle variazioni dei parametri di processo

3.1 Introduzione

3.2 Il diodo

- 3.2.1 Un primo sguardo al diodo: la regione di svuotamento
- 3.2.2 Il comportamento statico
- 3.2.3 Il comportamento dinamico o transitorio
- 3.2.4 Il diodo reale: effetti del secondo ordine
- 3.2.5 Il modello SPICE del diodo

3.3 Il transistor MOS(FET)

- 3.3.1 Un primo sguardo al dispositivo
- 3.3.2 Caratteristiche statiche del transistor MOS
- 3.3.3 Il comportamento dinamico del transistor MOS
- 3.3.4 Il transistor MOS reale: alcuni effetti secondari
- 3.3.5 Modelli SPICE per il transistor MOS

3.4 Una parola sulle variazioni del processo

3.5 Prospettive: evoluzione delle tecnologie

3.6 Riepilogo

3.7 Approfondimenti

Materiale protetto da copyright

3.1 Introduzione

Nel campo ingegneristico è ben noto che concepire una costruzione complessa senza una conoscenza preliminare degli elementi costruttivi di base porta a sicuro insuccesso. Senza dubbio questa considerazione è valida anche per la progettazione di circuiti digitali. Gli elementi costruttivi basilari nei circuiti digitali odierni sono i dispositivi a semiconduttore realizzati in silicio, più specificatamente i transistor MOS e, in grado minore, i diodi parassiti e le piste di interconnessione. Il ruolo dei dispositivi a semiconduttore è conosciuto da lungo tempo nel mondo dei circuiti integrati digitali, mentre le piste di interconnessione hanno cominciato solo recentemente a giocare un ruolo dominante a seguito dello scaling avanzato della tecnologia dei semiconduttori.

Fornire al lettore la *conoscenza* e la *comprensione* necessarie di questi componenti è la motivazione principale dei prossimi due capitoli. Non è nostra intenzione presentare un'analisi approfondita della fisica dei semiconduttori e delle piste di interconnessione. Per questo motivo indirizziamo il lettore ai molti libri eccellenti sui dispositivi a semiconduttore, alcuni dei quali sono citati alla fine del capitolo. Il nostro scopo è quello di dare una descrizione funzionale del modo di operare dei dispositivi, in modo da mettere in evidenza le proprietà e i parametri che sono di particolare importanza nella progettazione delle porte digitali e di introdurre alcune convenzioni sulla notazione.

Un'altra funzione importante di questo capitolo è di introdurre i *modelli* dei dispositivi. Tenere in considerazione tutti gli aspetti fisici di ogni singolo componente durante la progettazione di un sistema digitale complesso, comporta un'inutile complessità che diviene ben presto ingestibile. Un simile approccio corrisponderebbe a prendere in considerazione la struttura molecolare del cemento durante la costruzione di un ponte. Per affrontare tale problematica, si utilizza abitualmente un'astrazione o modello del comportamento del dispositivo. Per ciascun componente può essere concepita una varietà di modelli, a seconda del compromesso tra accuratezza e completezza. Un semplice modello del primo ordine è utile per un'analisi manuale. Ha un'accuratezza limitata, ma è di aiuto per comprendere il modo di operare del circuito e i suoi parametri dominanti. Nel caso in cui siano necessari risultati più accurati, è necessario usare modelli complessi del secondo ordine o anche di ordine maggiore con l'aiuto della simulazione numerica. In questo capitolo presentiamo sia i modelli del primo ordine per l'analisi manuale, sia i modelli di ordine più elevato per la simulazione di ciascun componente di interesse.

I progettisti tendono a dare per assodati i parametri dei componenti utilizzati nei modelli. Dovrebbero però essere consapevoli del fatto che si tratta solamente di valori nominali, e che i valori reali dei parametri non solo variano a seconda della temperatura, ma anche da un ciclo di produzione a un altro o addirittura nell'ambito di uno stesso wafer. Per evidenziare tale problematica, includiamo una breve discussione sulle variazioni di processo e sulle loro conseguenze.

Materiale protetto da copyright

3.2 Il diodo

Sebbene i diodi siano raramente inclusi direttamente negli schemi circuitali delle porte digitali attuali, essi sono ancora onnipresenti. Ciascun transistor MOS contiene implicitamente un certo numero di diodi polarizzati in zona inversa che influenzano direttamente il comportamento del dispositivo. In particolare, le capacità dipendenti dalla tensione applicata, associate a questi elementi parassiti, rivestono un ruolo importante nella commutazione di una porta digitale in tecnologia MOS. I diodi sono anche utilizzati per proteggere i dispositivi di ingresso di un circuito integrato dalle scariche elettrostatiche. Per queste ragioni, è appropriata una breve revisione delle proprietà fondamentali del diodo. Piuttosto che essere esaustivi, si preferisce focalizzarsi su quegli aspetti che condizionano la progettazione di circuiti digitali MOS, più precisamente il comportamento in zona di polarizzazione inversa[1]

3.2.1 Un primo sguardo al diodo: la regione di svuotamento

Il diodo a giunzione *pn* è il più semplice dei dispositivi a semiconduttore. La Figura 3-1 mostra la sezione di un tipica giunzione *pn*. Questa consiste in due regioni omogenee di materiale di tipo *p* e di tipo *n*, separate da una regione di transizione, che supponiamo molto sottile. Tale dispositivo è chiamato *giunzione a gradino* o *giunzione brusca*. Il materiale di tipo *p* è drogato con atomi *accettori* (ad es. boro), che introducono *lacune* come portatori maggioritari. Similmente, drogando il silicio con atomi *donori* (ad es. fosforo o arsenico), si crea una regione di tipo *n*, nella quale i portatori maggioritari sono gli elettroni. I contatti di alluminio permettono di accedere elettricamente ai terminali *p* e *n* del dispositivo. Il simbolo circuitale del diodo utilizzato abitualmente negli schemi circuitali è riportato in Figura 3-1c.

Per comprendere il comportamento del diodo *pn*, spesso facciamo ricorso ad una rappresentazione semplificata monodimensionale del dispositivo (Figura 3-1b). Accostare i materiali di tipo *p* e *n* produce un significativo gradiente della concentrazione di portatori nella regione di contatto: la concentrazione degli elettroni cambia da un valore elevato nel materiale di tipo *n* ad un valore molto inferiore nel materiale di tipo *p*. La situazione opposta si verifica nel caso della concentrazione delle lacune. Tale gradiente forza gli elettroni a *diffondere* dalla zona *n* a quella *p* e le lacune dalla zona *p* a quella *n*. Quando le lacune abbandonano il materiale di tipo *p*, lasciano degli ioni accettori che sono carichi negativamente. Di conseguenza, il materiale *p* risulta carico negativamente nelle vicinanze dell'interfaccia *p-n*. Similmente, un carica positiva si forma dalla parte di tipo *n* della giunzione, dato che gli elettroni, diffondendo, lasciano gli ioni donori, carichi positivamente. La regione della giunzione in cui sono venuti a mancare i portatori maggioritari, lasciando gli ioni fissi accettori e donori, è chiamata *regione di svuotamento* o *di carica spaziale*. Le cariche danno luogo ad un campo elettrico a cavallo dell'interfaccia che è diretto dalla regione *n* verso quella *p*. Tale campo si contrappone alla diffusione di lacune ed elettroni, poiché causa una deriva (*drift*) degli elettroni dalla zona *p* alla *n* e delle lacune dalla zona *n* alla *p*. In condizione di equilibrio, la carica fissa di svuotamento crea un campo elettrico

[1] Indirizziamo il lettore interessato al sito web del libro per una descrizione esaustiva del comportamento del diodo.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Esempio 3.1 Potenziale intrinseco in una giunzione *pn*

Una giunzione brusca ha una densità di drogaggio pari a $N_A = 10^{15}$ atomi/cm^3, e $N_D = 10^{16}$ atomi/cm^3. Calcolare il potenziale intrinseco a 300 K.

Dall'Eq. (3.1),

$$\phi_0 = 26 \ln\left[\frac{10^{15} \times 10^{16}}{2{,}25 \times 10^{20}}\right] \text{mV} = 638 \text{ mV}$$

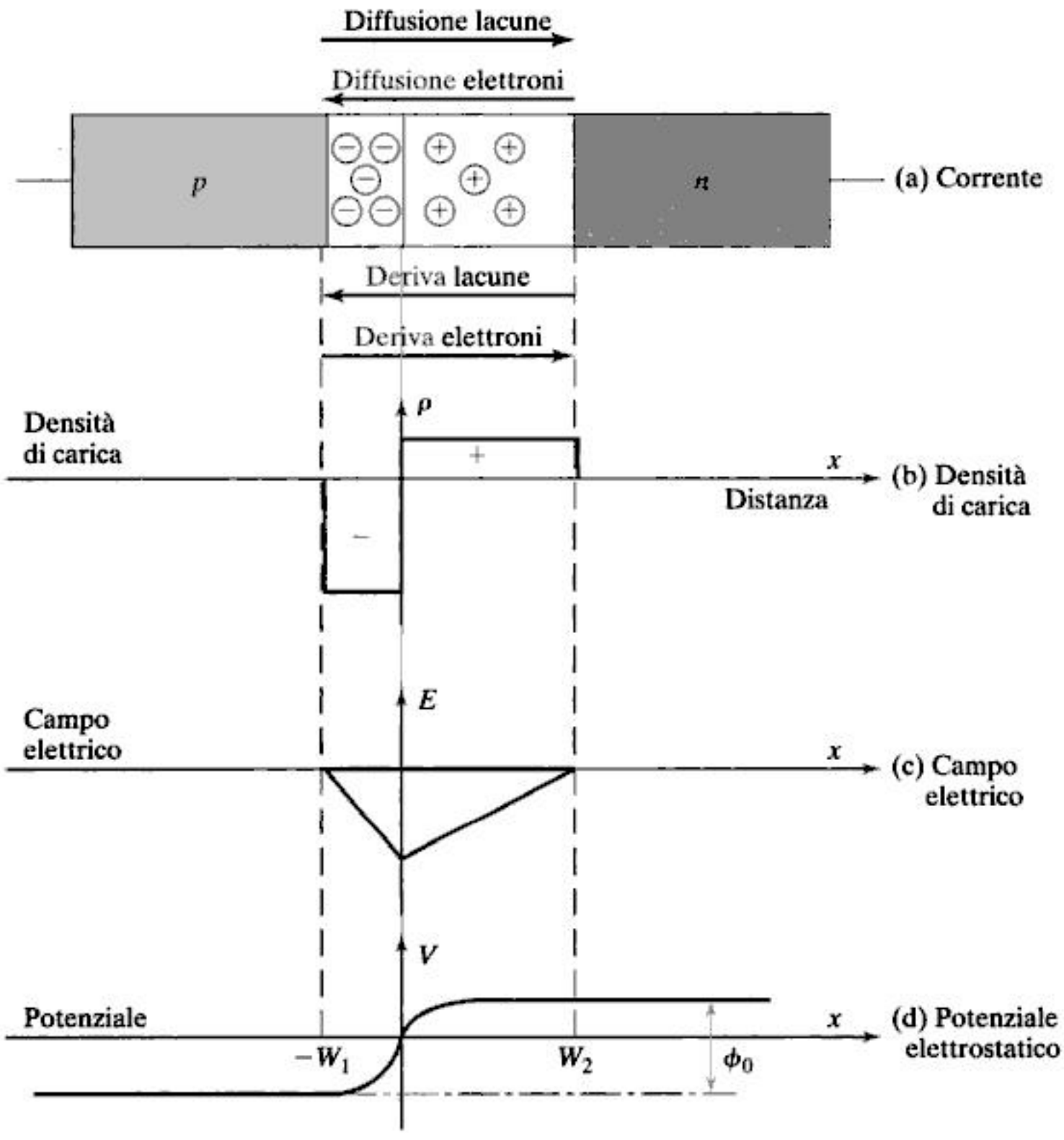


Figura 3-2 La giunzione brusca *pn* in condizione di equilibrio.

Materiale protetto da copyright

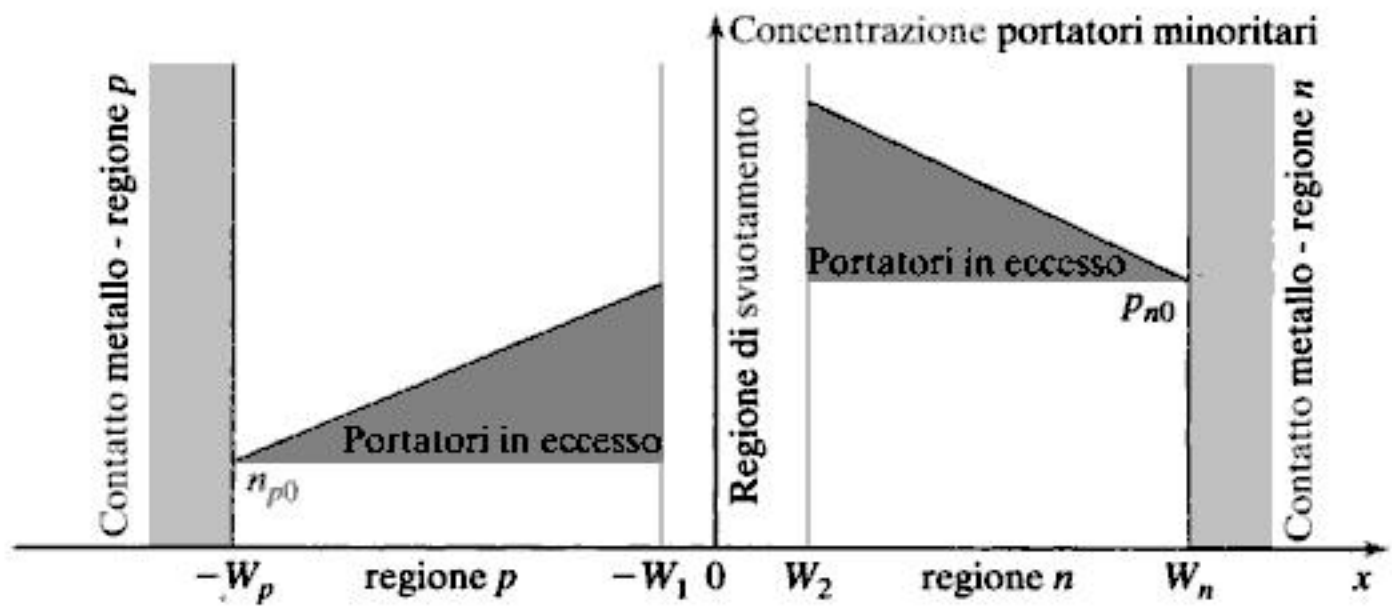


Figura 3-3 Concentrazione dei portatori minoritari nella regione neutra vicino a una giunzione *pn* brusca in condizione di polarizzazione diretta.

3.2.2 Il comportamento statico

Le equazioni del diodo ideale

Assumiamo ora che una tensione diretta V_D sia applicata alla giunzione: il potenziale della regione *p* risulta più elevato rispetto alla regione *n*, abbassando la barriera di potenziale rispetto al valore intrinseco. Di conseguenza, il flusso di portatori mobili attraverso la giunzione aumenta, poiché la componente di diffusione della corrente è dominante rispetto a quella di deriva. Questi portatori attraversano la regione di svuotamento e sono quindi iniettati nelle regioni neutre *n* e *p*, dove divengono portatori minoritari, come si vede in Figura 3-3. Nell'ipotesi che non esista alcun gradiente di potenziale nelle regioni neutre, come generalmente accade nei dispositivi moderni, questi portatori minoritari diffonderanno attraverso la regione a causa del gradiente di concentrazione, fino a ricombinarsi con un portatore maggioritario. Il risultato netto è un flusso di corrente attraverso il diodo, dalla regione *p* a quella *n* e il diodo è detto in *polarizzazione diretta*.

D'altra parte, quando si applica alla giunzione una tensione di polarizzazione inversa V_D (questo significa che il potenziale della regione *p* viene abbassato rispetto a quello della regione *n*) l'altezza della barriera di potenziale aumenta. Ciò comporta una riduzione della corrente di diffusione e quindi la corrente di deriva diviene dominante. La componente di deriva della corrente fluisce dalla regione *n* alla regione *p* ed è praticamente trascurabile, poiché la concentrazione di minoritari (elettroni nella zona *p*, lacune nella zona *n*) è molto ridotta (Figura 3-4). Verosimilmente si può affermare che in *polarizzazione inversa* il diodo si comporti come un dispositivo non-conduttivo o bloccante. Il diodo è quindi un conduttore monodirezionale.

La proprietà più importante della corrente del diodo è la *dipendenza esponenziale* dalla tensione di polarizzazione applicata. Tale andamento è mostrato in Figura 3-5, che riporta la corrente nel diodo I_D in funzione della tensione di polarizzazione applicata V_D. Il comportamento esponenziale nel caso di tensione di polarizzazione positiva è ancor più evidente in Figura 3-5b, dove la corrente è riportata in scala logaritmica. La corrente aumenta di un fattore 10 per ogni incremento di 60 mV (= 2.3 ϕ_T) della tensione di polarizzazione diretta. Per

Materiale protetto da copyright

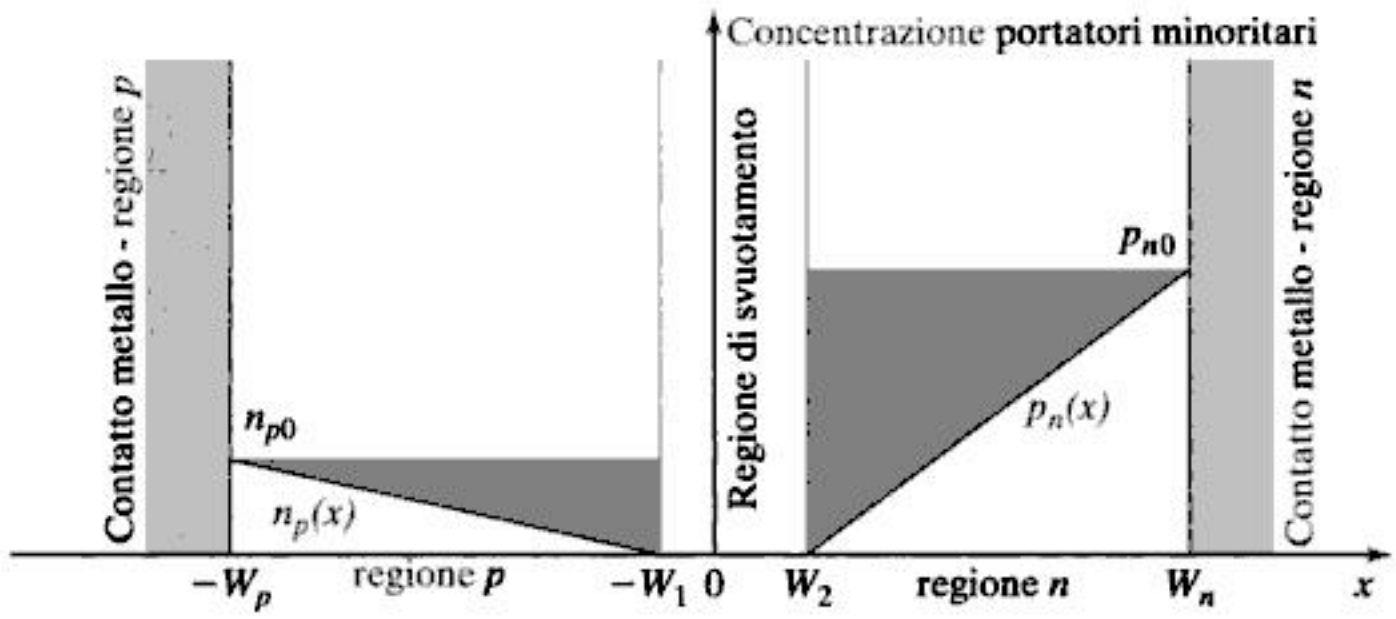


Figura 3-4 Concentrazione dei portatori minoritari nella regione neutra vicino alla giunzione *pn* in condizione di polarizzazione inversa.

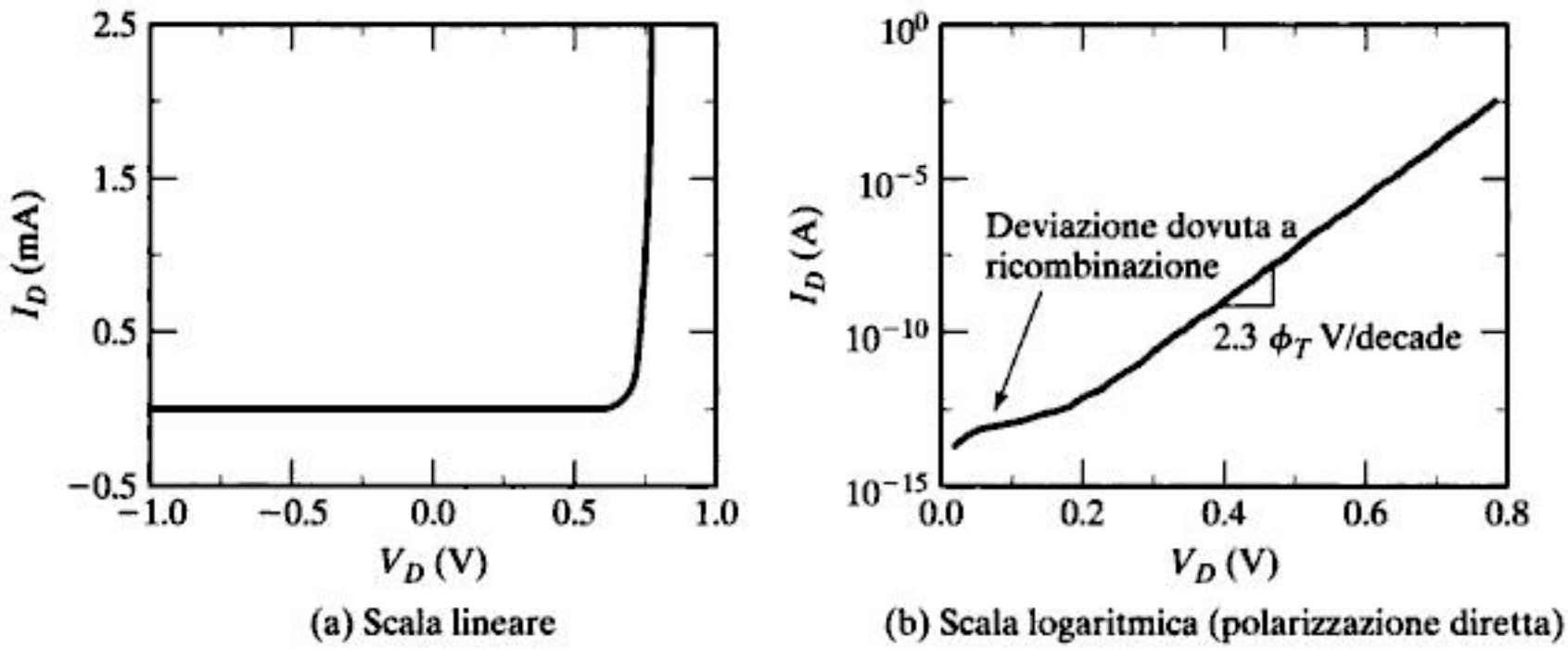


Figura 3-5 Corrente del diodo in funzione della tensione di polarizzazione V_D.

tensioni di polarizzazione ridotte ($V_D < 0.15$ V), si può notare una deviazione dal comportamento esponenziale, dovuto alla ricombinazione di lacune ed elettroni nella regione di svuotamento.

Il comportamento del diodo, sia in condizioni di polarizzazione diretta che inversa, è adeguatamente descritto dall'*equazione del diodo ideale*, che pone in relazione la corrente che scorre nel diodo I_D con la tensione di polarizzazione V_D:

$$I_D = I_S(e^{V_D/\phi_T} - 1) \tag{3.3}$$

Il comportamento esponenziale del diodo è mostrato in Figura 3-5, dove ϕ_T è il potenziale termico utilizzato nell'Eq. (3.2) ed è uguale a 26 mV a temperatura ambiente. I_S costituisce un termine costante, denominato *corrente di saturazione* del diodo. Questa è proporzionale all'area del diodo, ai livelli di drogaggio e alla profondità delle regioni neutre. Molto spesso il valore di I_S viene determinato sperimentalmente. Va specificato che nei dispositivi reali la corrente in regione inversa è significativamente maggiore della corrente di saturazione I_S. Questo fatto è dovuto alla generazione termica di coppie elettrone-lacuna nella regione di

Materiale protetto da copyright

svuotamento. Il campo elettrico presente in tale regione spinge tali portatori fuori dalla regione, causando una componente di corrente addizionale. Nel caso di giunzioni tipiche, la corrente di saturazione ha un valore nominale dell'ordine di 10^{-17} A/μm^2, mentre la corrente inversa totale è circa tre ordini di grandezza più elevata. Di conseguenza è necessario ricorrere a tecniche di misura sperimentale per determinare un valore attendibile delle correnti di perdita (*leakage*) attraverso diodi polarizzati in inversa.

Modelli per l'analisi manuale

L'equazione corrente-tensione ricavata può essere riassunta in un insieme di semplici modelli che sono utili per l'analisi manuale del comportamento del diodo. Uno di questo modelli, rappresentato in Figura 3-6a, si basa sull'equazione Eq. (3.3) ideale del diodo. Questo modello, pur portando a risultati molto accurati, ha lo svantaggio di essere fortemente non lineare, il che impedisce un'analisi rapida e del primo ordine per determinare le condizioni operative di un circuito in stato stazionario. Un modello semplificato utilizzato di frequente è ottenuto osservando l'andamento della corrente del diodo, come riportata in Figura 3-5. Per un diodo in "piena conduzione", la caduta di tensione ai suoi capi V_D è limitata in un intervallo ristretto, all'incirca tra 0.6 e 0.8 V. In un modello del primo ordine è quindi accettabile presumere che ai capi del diodo cada un tensione fissa V_{Don}. Sebbene il valore di V_{Don} in realtà dipenda da I_S, tipicamente si può assumere che sia costante e che valga 0.7 V, ottenendo quindi il modello di Figura 3-6b, in cui un diodo in conduzione è sostituito con un generatore di tensione costante, mentre un diodo interdetto è sostituito da un circuito aperto.

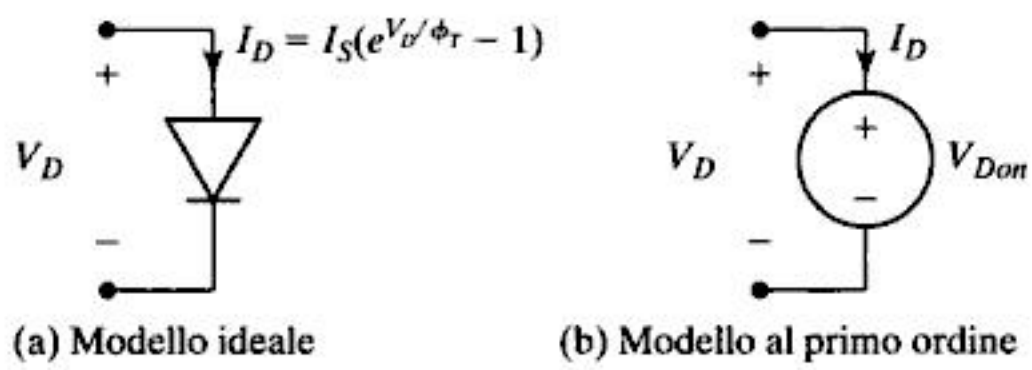


Figura 3-6 Modelli del diodo.

Esempio 3.2 Analisi di un circuito a diodi

Consideriamo la semplice rete di Figura 3-7 ed assumiamo che $V_S = 3$ V, $R_S = 10$ kΩ, e $I_S = 0.5 \times 10^{-16}$ A. La corrente e la tensione del diodo soddisfano la seguente equazione della maglia:

$$V_S - R_S I_D = V_D$$

Inserire l'equazione del diodo ideale e risolvere (faticosamente) l'equazione non lineare utilizzando metodi numerici o tecniche iterative porta alla seguente soluzione: $I_D = 0.224$ mA, e $V_D = 0.757$ V. Il modello semplificato con $V_{Don} = 0.7$ V porta a risultati del tutto simili ($V_D = 0.7$ V, $I_D = 0.23$ mA) ma con uno sforzo computazionale deci-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

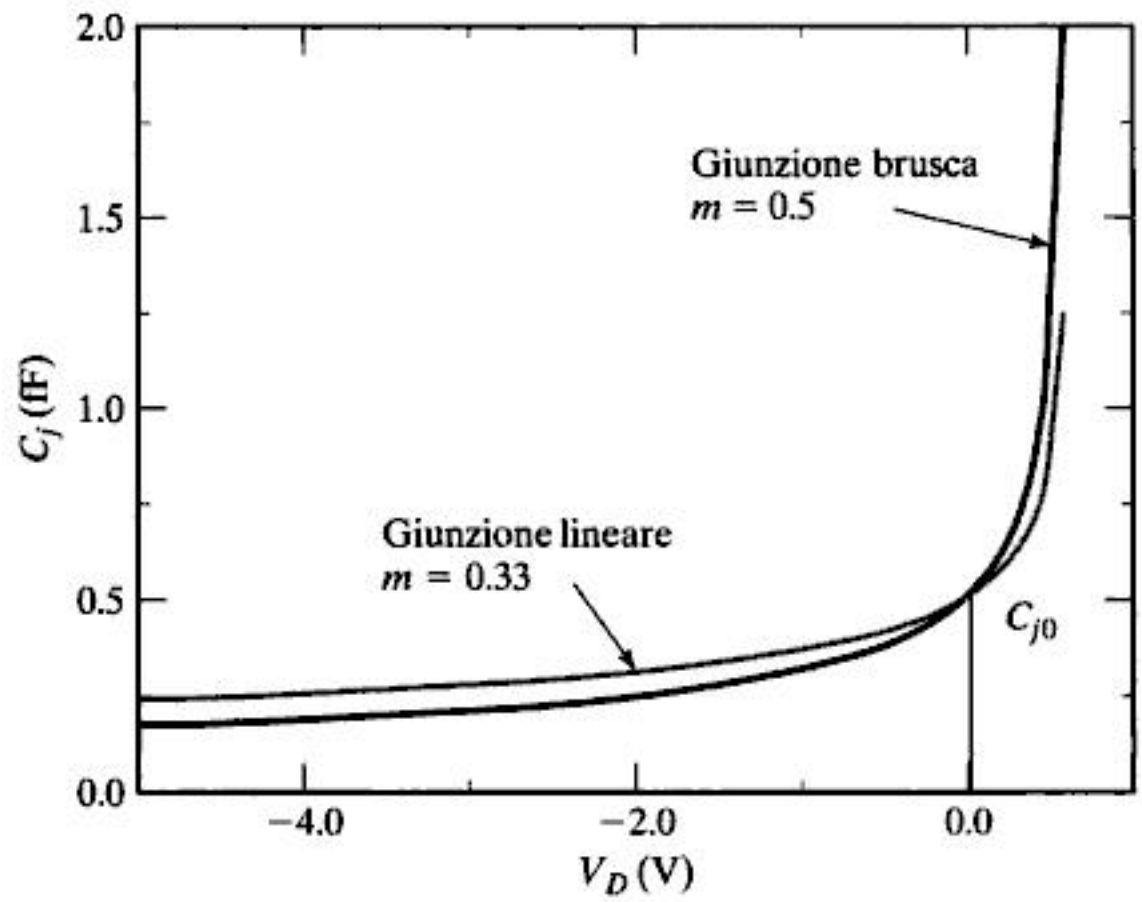


Figura 3-8 Capacità di giunzione (in fF/μm²) in funzione della tensione di polarizzazione applicata.

$$C_{j0} = A_D\sqrt{\left(\frac{\varepsilon_{si}q}{2}\frac{N_A N_D}{N_A + N_D}\right)\phi_0^{-1}} \tag{3.8}$$

Si noti che lo stesso valore di capacità si ottiene utilizzando la tipica equazione del condesatore a piatti paralleli $C_j = \varepsilon_{si} A_D/W_j$ (con W_j dato da Eq. (3.5)). Tipicamente si omette il fattore A_D e C_j e C_{j0} sono espresse come capacità per unità di area.

La risultante capacità di giunzione è riportata in funzione della tensione di polarizzazione in Figura 3-8 nel caso di un diodo che tipicamente si trova nei circuiti MOS. Una dipendenza *fortemente non lineare* è evidente. Si noti anche che la capacità diminuisce all'aumentare della polarizzazione inversa: una polarizzazione inversa di 5 V implica la riduzione della capacità di un fattore maggiore di 2.

Esempio 3.3 Capacità di giunzione

Consideriamo il seguente diodo a giunzione in silicio: $C_{j0} = 2 \times 10^{-3}$ F/m², $A_D = 0.5$ μm², e $\phi_0 = 0.64$ V. Una tensione inversa di −2.5 V comporta una capacità di giunzione pari a 0.9×10^{-3} F/m² (0.9 fF/μm²), o, per il diodo completo, una capacità di 0.45 fF.

L'Eq. (3.7) è valida solamente sotto la condizione che la giunzione *pn* sia una *giunzione brusca*, per la quale la transizione da materiale di tipo *n* a materiale di tipo *p* sia istantanea. Molto spesso questo non è il caso delle giunzioni *pn* in circuiti integrati reali, dove la transizione dal materiale *n* a quello *p* può essere graduale. In quei casi, una distribuzione lineare delle impurità a cavallo della giunzione rappresenta un'approssimazione migliore rispetto

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

2. Anche la corrente di saturazione I_S dipende dalla temperatura, poiché la concentrazione di portatori in equilibrio termico aumenta all'aumentare della temperatura. In teoria la corrente di saturazione dovrebbe circa raddoppiare ogni 5°C. Sperimentalmente si è misurato il raddoppio della corrente ogni 8°C.

Questa doppia dipendenza ha un impatto significativo sul funzionamento di un circuito digitale. In primo luogo, i livelli di corrente (e quindi il consumo di potenza) possono aumentare in modo sostanziale. Con una tensione di polarizzazione diretta pari a 0.7 V a 300°K, per esempio, la corrente aumenta circa del 6% per ogni °C, e raddoppia circa ogni 12°C. A parità di corrente, la tensione del diodo V_D diminuisce di alcuni mV ogni °C. In secondo luogo, i circuiti integrati fanno molto affidamento sui diodi polarizzati in inversa come dispositivi idealmente isolanti: un aumento della temperatura comporta l'aumento delle correnti di perdita e quindi il deterioramento delle proprietà di isolante.

3.2.5 Il modello SPICE del diodo

Nei paragrafi precedenti abbiamo presentato un modello per l'analisi manuale dei circuiti a diodi. Nel caso di circuiti più complessi o in caso sia necessario utilizzare un modello più complesso che includa anche effetti del secondo ordine, l'analisi manuale diviene impraticabile ed è quindi necessario utilizzare la simulazione assistita al calcolatore. Sebbene negli ultimi decenni siano stati sviluppati diversi simulatori circuitali, il programma SPICE, sviluppato presso il Campus di Berkeley della University of California, è decisamente quello di maggior successo [Nagel75]. La simulazione di un circuito integrato che contiene dei dispositivi attivi necessita di un modello matematico per quei dispositivi, che da qui in poi chiameremo il *modello SPICE*. L'accuratezza della simulazione dipende direttamente dalla qualità di tale modello. Per esempio non ci si può aspettare di vedere nella simulazione i risultati di effetti del secondo ordine, se tali effetti non sono inclusi nel modello del dispositivo. Creare modelli accurati ed al contempo efficienti dal punto di vista computazionale è stato un processo lungo e tedioso che non è per nulla terminato. Nel tempo ciascuno dei maggiori produttori di semiconduttori ha sviluppato i propri modelli proprietari, pretendendo di aver ottenuto la migliore accuratezza o la migliore efficienza computazionale e robustezza. Fortunatamente negli ultimi tempi si è osservato un certo consolidamento nel campo di questi modelli.

Il modello SPICE standard per un diodo è semplice, come mostrato in Figura 3-10. La caratteristica del diodo in stato stazionario è modellizzata dal generatore di corrente non lineare I_D, che è una versione modificata dell'equazione ideale del diodo:

$$I_D = I_S(e^{V_D/n\phi_T} - 1) \tag{3.12}$$

Il parametro aggiuntivo n è detto *coefficiente di idealità*. Tale parametro è uguale a 1 per i diodi più comuni, ma può anche essere in certa misura maggiore di 1 per altri diodi. La resistenza R_s modella la resistenza serie dovuta alla regioni neutre ad entrambi i lati della giunzione. Per alti livelli di corrente, a causa di questa resistenza il potenziale interno del

Materiale protetto da copyright

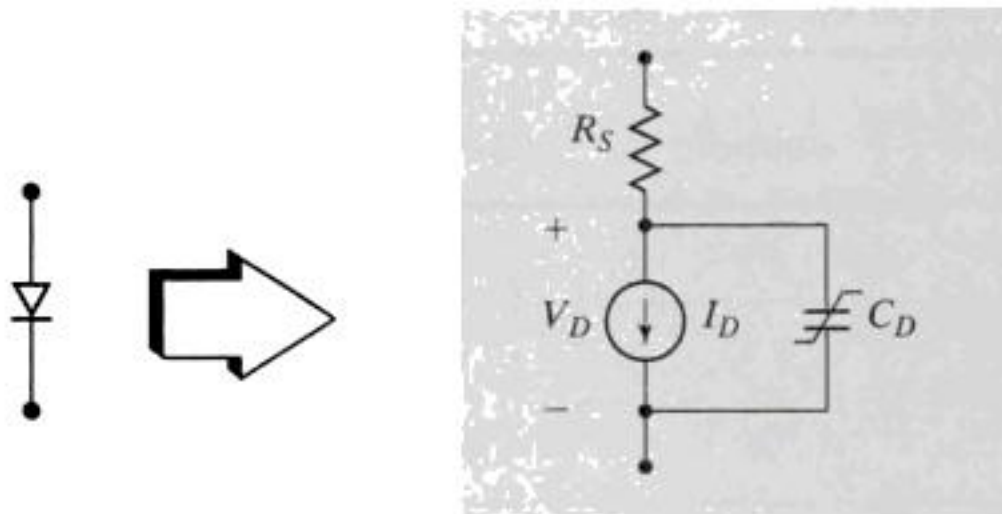


Figura 3-10 Modello SPICE del diodo.

diodo V_D differisce da quello applicato esternamente, e la corrente risulta quindi meno elevata rispetto a quanto atteso in accordo all'equazione ideale del diodo.

Il comportamento dinamico del diodo è modellizzato dalla capacità non lineare C_D, che tiene conto dei due fenomeni di immagazzinamento di carica nel diodo: la carica spaziale (o della regione di svuotamento) e la carica associata all'eccesso di portatori minoritari. In questo capitolo abbiamo descritto solo il primo fenomeno, poiché il secondo è significativo solo in condizioni di polarizzazione diretta. Il modello per la capacità non lineare è definito di seguito:

$$C_D = \frac{C_{j0}}{(1 - V_D/\phi_0)^m} + \frac{\tau_T I_S}{\phi_T} e^{V_D/n\phi_T} \tag{3.13}$$

Una lista dei parametri utilizzati nel modello del diodo è riportata in Tabella 3-1. Oltre al nome del parametro, il simbolo ed il nome utilizzato in SPICE, la tabella contiene anche il valore predefinito utilizzato in SPICE nel caso il parametro non venga specificato. Questa tabella non è in nessun modo completa: altri parametri sono disponibili per includere altri effetti del secondo ordine quali breakdown, alti livelli di iniezione e rumore. Per brevità, ci siamo limitati ad elencare i parametri che sono di interesse specifico per questo libro. Per una descrizione completa dei modelli dei dispositivi e dell'uso di SPICE, rinviamo il lettore ai numerosi libri di testo sull'argomento (vedi ad esempio [Banhzaf92], [Thorpe92], e [Vladimirescu93]).

3.3 Il transistor MOS(FET)

Il transistor ad effetto di campo metallo-ossido-semiconduttore (*metal-oxide-semiconductor*, *MOSFET* o in breve *MOS*) è certamente il cavallo di battaglia dei sistemi digitali contemporanei. Il suo vantaggio principale, nell'ottica di sistemi digitali, sta nel fatto che si comporta bene come interruttore, introducendo pochi elementi parassiti. Altri vantaggi principali sono la densità di integrazione e un processo di fabbricazione relativamente semplice, che hanno permesso la produzione a costi ridotti di sistemi complessi e con un numero elevato di transistor.

Materiale protetto da copyright

Tabella 3-1 Parametri del modello SPICE del primo ordine del diodo.

Nome del parametro	Simbolo	Nome in SPICE	Unità	Valore predefinito
Corrente di saturazione	I_S	IS	A	1.0 E–14
Coefficiente di idealità	n	N	–	1
Resistenza serie	R_S	RS	Ω	0
Tempo di transizione	τ_T	TT	s	0
Capacità di giunzione per polarizzazione nulla	C_{j0}	CJ0	F	0
Coefficiente di profilo	m	M	–	0.5
Potenziale di giunzione	ϕ_0	VJ	V	1

Seguendo l'approccio utilizzato nel caso del diodo, in questo paragrafo limitiamo la trattazione a una panoramica generale del transistor e dei suoi parametri fondamentali. Dopo tale panoramica generale, presenteremo una descrizione analitica del transistor, sia dal punto di vista statico (stato stazionario) che dinamico (transitorio). Concluderemo l'analisi con un elenco di alcuni effetti del secondo ordine e con l'introduzione di modelli SPICE del transistor MOS.

3.3.1 Un primo sguardo al dispositivo

Il MOSFET è un dispositivo a quattro terminali. La tensione applicata al terminale di *gate* determina se e quanta corrente scorre tra i poli di *source* e *drain*. Il *substrato* rappresenta il quarto terminale del transistor. La sua funzione è secondaria in quanto ha l'unico effetto di modulare le caratteristiche e i parametri del dispositivo.

A livello di astrazione più elevato, il transistor può essere visto come un interruttore. Quando viene applicata al gate una tensione maggiore di un determinato valore critico (la *tensione di soglia* V_T), si forma un canale conduttivo tra drain e source. In presenza di una differenza di potenziale tra questi ultimi si ha quindi un flusso di corrente. La conducibilità del canale è modulata dalla tensione di gate: maggiore è la differenza di tensione tra gate e source, minore è la resistenza del canale conduttivo e quindi maggiore è la corrente. Quando la tensione di gate è minore della tensione di soglia, il canale conduttivo non è presente e quindi l'interruttore si può considerare aperto.

Si possono distinguere due tipi di MOSFET. Il transistor NMOS è formato da regioni di drain e source di tipo n^+ circondate da un substrato di tipo p. La corrente è costituita dagli elettroni che si muovono attraverso il canale di tipo n, dal source al drain, al contrario di quanto accade nel diodo a giunzione pn, dove al flusso di corrente contribuiscono sia gli elet-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

All'aumentare della tensione di gate, il potenziale alla superficie del silicio raggiunge, ad un certo punto, un valore critico per cui la superficie del semiconduttore si inverte e diviene di tipo *n*. Questo punto segna l'inizio di un fenomeno conosciuto come *forte inversione* (*strong inversion*), che si manifesta ad una tensione pari a due volte il *potenziale di Fermi* (Eq. (3.16)) ($\phi_F \approx -0.3$ V per tipici substrati in silicio di tipo *p*):

$$\phi_F = \phi_T \ln\left(\frac{N_A}{n_i}\right) \tag{3.16}$$

Un ulteriore aumento della tensione di gate non comporta cambiamenti ulteriori nella regione di svuotamento, ma fa aumentare il numero di elettroni nel sottile strato di inversione direttamente al di sotto dell'ossido. Questi ultimi sono trasportati dalla regione di source, fortemente drogata *n*+, allo strato di inversione. Quindi si è formato un canale continuo di tipo *n* tra le regioni di source e drain, la cui conducibilità è modulata dalla tensione gate-source.

In presenza dello strato di inversione, la carica immagazzinata nella regione di svuotamento è fissa e vale

$$Q_{B0} = \sqrt{2qN_A\varepsilon_{si}|2\phi_F|} \tag{3.17}$$

La situazione cambia quando si applica una tensione di polarizzazione V_{SB} tra source e substrato (V_{SB} normalmente è positiva per dispositivi a canale *n*). Questo fatto fa sì che il potenziale superficiale richiesto per instaurare la forte inversione aumenti e divenga $|-2\phi_F + V_{SB}|$. La carica immagazzinata nella regione di svuotamento può quindi essere espressa come

$$Q_B = \sqrt{2qN_A\varepsilon_{si}(|(-2)\phi_F + V_{SB}|)}\,. \tag{3.18}$$

Il valore di V_{GS} per cui si instaura la forte inversione è detto *tensione di soglia* (*threshold voltage*, V_T). V_T è funzione di molti fattori, molti dei quali sono costanti tipiche del materiale, quali la differenza tra le funzioni lavoro del materiale del gate e del substrato, lo spessore dell'ossido, il potenziale di Fermi, la carica delle impurità intrappolate alla superficie tra canale ed ossido di gate e la dose di ioni impiantati per l'aggiustamento della soglia.

Dalla trattazione fin qui svolta dovrebbe risultare evidente che la tensione source-bulk V_{SB} influenza a sua volta la tensione di soglia. Piuttosto che fare affidamento su un'espressione analitica complessa (e tra l'altro poco accurata) per la tensione di soglia, sfruttiamo un parametro empirico V_{T0}, che equivale alla tensione di soglia per $V_{SB} = 0$, ed è principalmente funzione del processo di fabbricazione. La tensione di soglia per differenti condizioni di polarizzazione del substrato può quindi essere ricavata come

$$V_T = V_{T0} + \gamma(\sqrt{|(-2)\phi_F + V_{SB}|} - \sqrt{|2\phi_F|}) \tag{3.19}$$

Il parametro γ è chiamato *coefficiente dell'effetto body* e quantifica l'effetto delle variazioni di V_{SB}. Si osservi che la tensione di soglia ha un valore **positivo** per un tipico transistor **NMOS**, mentre è **negativa** per un normale transistor **PMOS**.

In Figura 3-14 è riportato l'effetto della polarizzazione del substrato sulla tensione di soglia di un transistor NMOS, assumendo $|-2\phi_F| = 0.6$ V e $\gamma = 0.4$ $V^{0.5}$. Una polarizzazione

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

rimane costante a $V_{GS} - V_T$ e, di conseguenza, la corrente rimane costante (in altre parole satura). Sostituendo V_{DS} con $V_{GS} - V_T$ nell'Eq. (3.25) si ottiene la corrente di drain in regime di saturazione. Vale la pena osservare che, in prima approssimazione, la corrente non è più funzione di V_{DS}; si noti anche la *dipendenza quadratica* della corrente di drain dalla tensione di controllo V_{GS}:

$$I_D = \frac{k_n'}{2}\frac{W}{L}(V_{GS} - V_T)^2 \tag{3.29}$$

Modulazione della lunghezza di canale

L'Eq. (3.29) descrive il transistor in saturazione come un generatore di corrente ideale: la corrente tra i terminali di drain e source è costante e indipendente dalla tensione applicata tra i due terminali. Tale scenario non è in realtà completo, poiché la lunghezza di canale effettiva è in realtà modulata dalla tensione applicata V_{DS}: l'aumento di V_{DS} causa l'allargamento della regione di svuotamento alla giunzione di drain con conseguente diminuzione della lunghezza di canale effettiva. Come si osserva nell'Eq. (3.29), la corrente aumenta quando la lunghezza L si riduce. Un descrizione più accurata della corrente nel transistor MOS è quindi data da

$$I_D = I_D'(1 + \lambda V_{DS}) \tag{3.30}$$

dove I_D' è la corrente prevista dall'espressione ricavata in precedenza (cioè, Eq. (3.29)) e λ è un parametro empirico chiamato *modulazione della lunghezza di canale*. Le espressioni analitiche di λ si sono sempre dimostrate complesse e poco accurate. In generale, λ è proporzionale all'inverso della lunghezza di canale. In transistor con canale corto, la regione di svuotamento della giunzione drain-substrato rappresenta una frazione significativa dell'intero canale e quindi l'effetto di modulazione del canale è più pronunciato. Risulta quindi consigliabile utilizzare transistor a canale lungo ogni qualvolta sia necessario un generatore di corrente con impedenza di uscita elevata.

Saturazione di velocità

Il comportamento di transistor con una lunghezza di canale molto ridotta (detti *dispositivi a canale corto*) è sensibilmente diverso dai modelli resistivo e di saturazione presentati fin qui. L'imputato principale per tale deviazione è l'effetto di *saturazione della velocità*. In accordo all'Eq. (3.23), la velocità dei portatori è proporzionale al campo elettrico, indipendentemente dal valore che tale campo assume; in altre parole la mobilità dei portatori risulta costante. Purtroppo per valori elevati del campo elettrico (longitudinale), i portatori non seguono più il modello lineare. Di fatto, quando il campo elettrico lungo il canale raggiunge un valore critico ξ_c, la velocità dei portatori tende a saturare a causa delle collisioni tra i portatori (*scattering*). Tale situazione è mostrata in Figura 3-17.

La velocità di saturazione per elettroni e lacune è all'incirca la stessa: 10^5 m/s. Il valore critico del campo elettrico per cui si raggiunge tale saturazione dipende dai livelli di drogaggio e dal campo elettrico trasversale (attraverso l'ossido). Nel caso degli elettroni tale valore può variare tra 1 e 5 V/μm. Questo significa che nel caso di un dispositivo NMOS con una lunghezza di canale pari a 0.25 μm, bastano solamente un paio di Volt applicati tra drain e

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

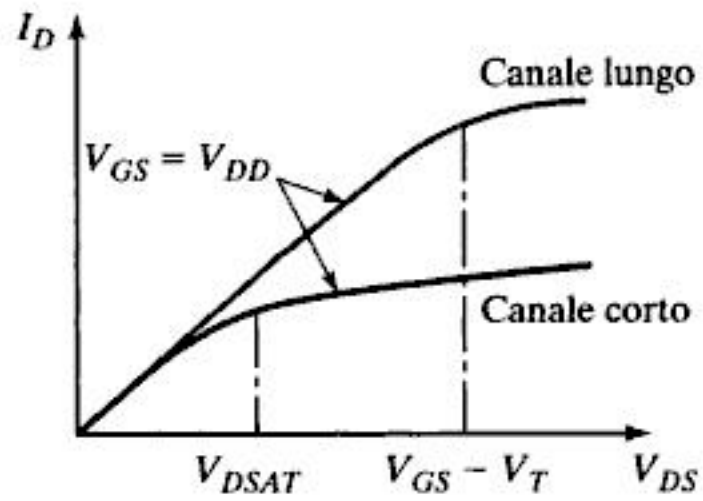


Figura 3-18 I dispositivi a canale corto mostrano una regione di saturazione più estesa a causa della saturazione di velocità.

Incrementando la tensione tra drain e source, il campo elettrico ad un certo punto supererà necessariamente il valore critico e quindi i portatori satureranno in velocità. Il valore della tensione di drain per cui si verifica la saturazione V_{DSAT} può essere calcolato uguagliando la corrente di saturazione alla corrente prevista dall'Eq. (3.32) per $V_{DS} = V_{DSAT}$ [Toh88]. La corrente di saturazione si può ricavare dall'Eq. (3.22), ipotizzando che la velocità di deriva abbia raggiunto il suo valore massimo υ_{sat}. Si ottiene quindi

$$\begin{aligned} I_{DSAT} &= \upsilon_{sat} C_{ox} W (V_{GT} - V_{DSAT}) \\ &= \kappa(V_{DSAT}) \mu_n C_{ox} \frac{W}{L} \left[V_{GT} V_{DSAT} - \frac{V_{DSAT}^2}{2} \right] \end{aligned} \tag{3.34}$$

dove V_{GT} è una notazione abbreviata per $V_{GS} - V_T$. Risolvendo l'equazione si ottiene

$$V_{DSAT} = \kappa(V_{GT}) V_{GT} \tag{3.35}$$

Un ulteriore incremento della tensione tra drain e source non comporta nessun aumento della corrente (almeno al primo ordine) e quindi la corrente del transistor satura a I_{DSAT}. Questo fatto suggerisce due osservazioni:

- Per un transistor a canale corto e per valori sufficientemente elevati di V_{GT}, $\kappa(V_{GT})$ è significativamente minore di 1 e $V_{DSAT} < V_{GT}$. Il dispositivo quindi entra in saturazione prima che il valore di V_{DS} raggiunga $V_{GS} - V_T$. Pertanto i dispositivi a canale corto hanno una regione di saturazione più estesa, tendono cioè ad operare maggiormente in condizioni di saturazione, rispetto ai dispositivi a canale lungo (vedi Figura 3-18).
- La corrente di saturazione I_{DSAT} *dipende linearmente* dalla tensione gate-source V_{GS}, al contrario dei dispositivi a canale lungo dove la dipendenza è quadratica. Questo riduce il livello di corrente massima che un transistor può erogare per una data tensione di controllo. D'altra parte, significa anche che la riduzione delle tensioni operative non ha un impatto altrettanto significativo a quello che si verifica nei dispositivi a canale lungo.

Le equazioni discusse fin qui trascurano il fatto che, a seguito di un aumento ulteriore di V_{DS}, una porzione maggiore del canale risulta saturata in velocità. Dal punto di vista della modella-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

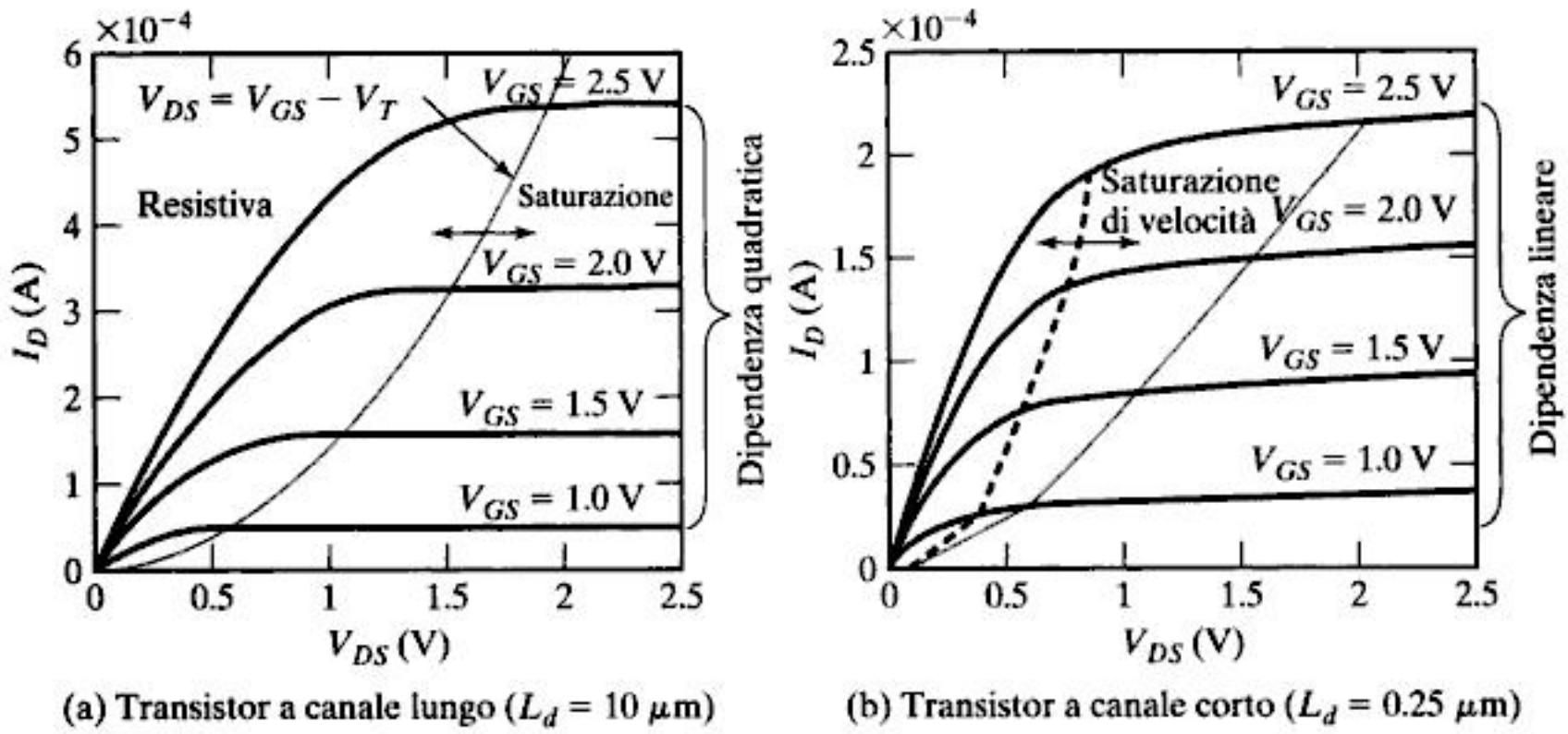


Figura 3-19 Caratteristica *I-V* di transistor NMOS a canale lungo ed a canale corto in una tecnologia CMOS da 0.25 μm. Il rapporto (*W/L*) di entrambi i transistor è identico ed uguale a 1.5. Si osservi la differenza nella scala dell'asse *y*.

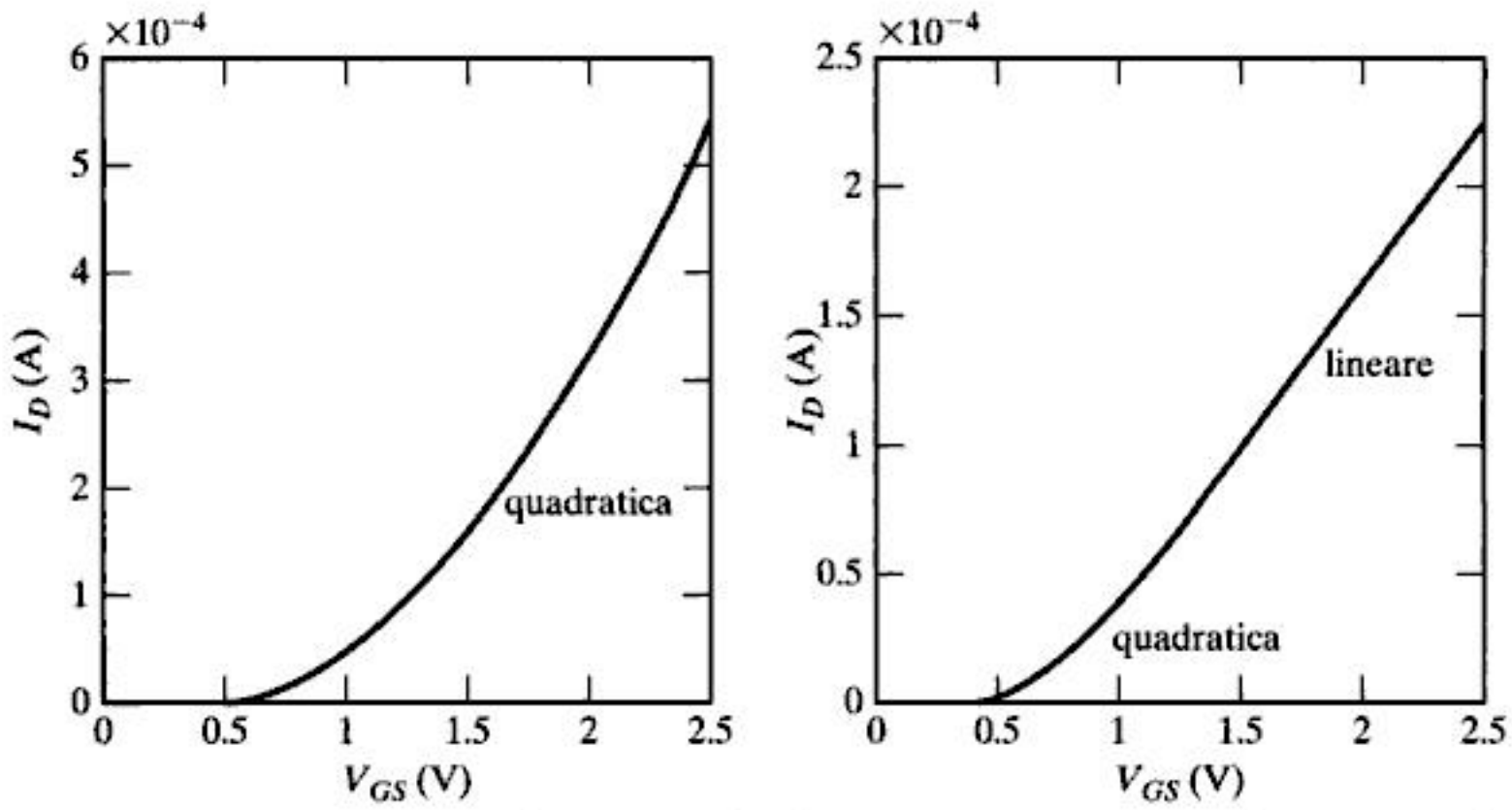
parametri del modello, è ancora possibile ottenere una corrispondenza decente con i dati sperimentali, come sarà dimostrato più avanti. Ancor più rilevante è il fatto che queste equazioni sono coerenti con il modello classico a canale lungo e quindi forniscono al progettista digitale un utilissimo strumento per la comprensione e l'interpretazione intuitive del circuito.

Caratteristiche I_D-V_{DS} e I_D-V_{GS}

Il modo migliore per comprendere il modo di operare del transistor MOS è quello di osservare le caratteristiche I_D-V_{DS}, che riportano I_D in funzione di V_{DS} assumendo V_{GS} come parametro. La Figura 3-19 riporta tali grafici per due transistor NMOS, realizzati con la stessa tecnologia e con lo stesso rapporto W/L. Ci si aspetta quindi che i due dispositivi abbiano la stessa caratteristica *I-V*; la differenza principale è però che il primo dispositivo è a canale lungo ($L_d = 10$ μm), mentre il secondo è un dispositivo a canale corto ($L_d = 0.25$ μm) e quindi soggetto alla saturazione di velocità.

Prendiamo dapprima in considerazione il dispositivo a canale lungo. Nella regione resistiva il transistor di comporta in modo equivalente a una resistenza controllata in tensione, mentre in regione di saturazione si comporta come un generatore di corrente controllato in tensione (ignorando l'effetto della modulazione della lunghezza di canale). La transizione tra le due regioni di funzionamento è delineata dalla curva $V_{DS} = V_{GS} - V_T$. La dipendenza quadratica di I_D da V_{GS} nella regione di saturazione (tipica dei dispositivi a canale lungo) è evidente nella distanza tra le varie caratteristiche. La dipendenza lineare della corrente di saturazione da V_{GS} è invece evidente per il dispositivo a canale corto di Figura 3-19b. Si può anche notare come la saturazione di velocità fa sì che la corrente del dispositivo saturi per un valore significativamente più ridotto di V_{DS}. La linea tratteggiata mostra i valori di tensione ai quali si verifica la saturazione di velocità, così come previsto dall'Eq. (3.35), mentre la linea a tratto continuo

Materiale protetto da copyright



(a) Transistor a canale lungo (L_d = 10 μm) (b) Transistor a canale corto (L_d = 0.25 μm)

Figura 3-20 Caratteristica $I_D - V_{GS}$ di transistor NMOS a canale lungo e a canale corto (tecnologia CMOS da 0.25 µm). W/L = 1.5 per entrambi i transistor e V_{DS} = 2.5 V.

mostra il valore per cui si sarebbe verificata la saturazione tradizionale ($V_{DS} = V_{GT}$). Questo fenomeno comporta una riduzione significativa della capacità di erogare corrente per valori elevati di tensione. Per esempio, per V_{GS} = 2.5 V e V_{DS} = 2.5 V, la corrente di drain del transistor corto è solamente il 40% di quella del dispositivo lungo (220 µA contro 540 µA).

La diversa dipendenza da V_{GS} tra dispositivi a canale lungo o corto è ancor più evidente in un altro insieme di grafici simulati, che riportano I_D in funzione di V_{GS} per un dato valore costante di V_{DS}, tale da assicurare che entrambi i dispositivi siano in saturazione (vedi Figura 3-20). Per valori elevati di V_{GS} è evidente la differenza tra la dipendenza quadratica e quella lineare.

Tutte le equazioni derivate fin qui sono valide anche per i transistor PMOS. L'unica differenza è che **per i dispositivi PMOS, la polarità di tutte le tensioni e correnti è invertita**. Questo è mostrato in Figura 3-21, che riporta la caratteristica I_D-V_{DS} di un transistor PMOS a dimensione minima nel nostro processo CMOS generico da 0.25 µm. Le curve sono nel terzo quadrante del piano cartesiano, poiché I_D, V_{DS}, e V_{GS} sono tutte negative. Si noti che gli effetti della saturazione di velocità sono meno pronunciati rispetto al caso dei dispositivi NMOS. Questo fatto è attribuibile al maggior valore del campo elettrico critico, dovuto alla minore mobilità delle lacune rispetto a quella degli elettroni.

La regione di sottosoglia

Osservando più attentamente le curve I_D-V_{GS} della Figura 3-20 si può notare che la corrente non crolla bruscamente a 0 per $V_{GS} = V_T$. Diviene quindi evidente che il transistor MOS in effetti è già parzialmente conduttivo per tensioni minori della tensione di soglia. Questo effetto è chiamato conduzione in *sottosoglia* o *inversione debole* (*weak-inversion*). L'inizio dell'inversione forte è caratterizzato da un'ampia disponibilità di portatori per la conduzione,

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

effetti le regioni di source (n^+) – substrato (p) – drain (n^+) formano un transistor bipolare parassita. La corrente in tale regione può essere approssimata dalla relazione seguente:

$$I_D = I_S e^{\frac{V_{GS}}{nkT/q}}\left(1 - e^{-\frac{V_{DS}}{kT/q}}\right)(1 + \lambda V_{DS}) \tag{3.39}$$

dove I_S e n sono parametri empirici, con $n \geq 1$, e tipicamente pari circa a 1.5.

Nella maggior parte delle applicazioni digitali, la corrente di sottosoglia è svantaggiosa, poiché altera la corrispondenza tra il transistor MOS e l'interruttore, spesso sfruttata nei circuiti logici. Sarebbe quindi preferibile avere un decadimento della corrente più rapido possibile non appena la tensione gate-source scenda al di sotto di V_T. Il tasso di riduzione della corrente rispetto a V_{GS} (in sottosoglia) è quindi una figura di merito del dispositivo ed è spesso quantificato mediante il valore S della pendenza espressa in mV/decade. Questa è una misura di quanto sia necessario ridurre V_{GS} perché la corrente diminuisca di un fattore 10. Dall'Eq. (3.39) troviamo quindi

$$S = n\left(\frac{kT}{q}\right)\ln(10) \tag{3.40}$$

dove S è espresso mV/decade. In un transistor ideale $n = 1$ e $(kT/q)\ln(10)$ vale 60 mV/decade a temperatura ambiente, il che significa che la corrente di sotto soglia diminuisce di un fattore 10 per una riduzione di V_{GS} pari a 60 mV. Sfortunatamente n è maggiore di 1 nei dispositivi reali e quindi la corrente decade con un tasso minore (90 mV/decade per $n = 1.5$). Inoltre il decadimento è ulteriormente attenuato dall'aumento della temperatura: la maggior parte dei circuiti integrati opera ad una temperatura sensibilmente maggiore di quella ambiente. Il valore di n dipende dalla topologia e dalla struttura intrinseche del dispositivo, quindi ridurre tale valore significa ricorrere ad una differente tecnologia, come per esempio il transistor SOI (*silicon on insulator*).

La corrente di sottosoglia ha alcune importanti conseguenze. In linea generale, vogliamo che la corrente sia il più possibile vicino a zero quando $V_{GS} = 0$. Questo è particolarmente importante nei *circuiti dinamici*, i quali si basano sulla possibilità di immagazzinare carica su di un condensatore e il cui funzionamento può essere seriamente compromesso dalle correnti di sottosoglia. Diviene quindi necessario definire un livello inferiore alla tensione di soglia dei dispositivi.

Esempio 3.6 Pendenza della corrente di sottosoglia

Per l'esempio di Figura 3-22, si osserva una pendenza di 89.5 mV/decade (tra 0.2 e 0.4 V). Questo equivale ad un fattore n pari a 1.49.

Materiale protetto da copyright

Modelli del transistor MOS

La trattazione fin qui svolta ha messo in chiara evidenza che il transistor sub-micrometrico è un dispositivo complesso. Il suo comportamento è certamente non lineare ed è influenzato da molti effetti del secondo ordine. Fortunatamente, come mostreremo più avanti, sono stati sviluppati dei modelli per la simulazione circuitale che permettono di predire il comportamento di un dispositivo con sorprendente precisione per un ampio intervallo di dimensioni e geometrie del dispositivo e in differenti condizioni operative. Sebbene questi modelli siano perfetti dal punto di vista della precisione, non permettono al progettista di avere una comprensione intuitiva del comportamento di un circuito e dei parametri dominanti nel sistema. Tale comprensione è però fondamentale sia in fase si analisi che di ottimizzazione del sistema. Al progettista che manca di una visione chiara di ciò che sovraintende e domina il comportamento di in circuito, non rimane che affidarsi ad un lungo processo di ottimizzazione per tentativi, che però in linea generale porta sempre ad una soluzione non ottimale.

Il punto cruciale diviene quindi capire come traslare la descrizione del comportamento complesso del nostro transistor MOS in un modello analitico semplificato e gestibile che non ci conduca ad equazioni complesse e insormontabili, pur non mancando di cogliere tutti gli aspetti essenziali del dispositivo. Si scopre in realtà che le equazioni del primo ordine ricavate in precedenza, possono essere combinate in un'unica espressione che soddisfa a tutti questi requisiti. Il modello considera il transistor come un unico generatore di corrente (come mostrato in Figura 3-23), il cui valore è esplicitato nella figura. Il lettore può verificare che, a seconda della condizione operativa, il modello corrisponde all'Eq. (3.25), all'Eq. (3.29) o all'Eq. (3.38) (includendo il termine correttivo per la modulazione della lunghezza di canale) [5]. Infine otteniamo:

$$I_D = 0 \text{ per } V_{GT} \leq 0$$

$$I_D = k'\frac{W}{L}\left(V_{GT}V_{min} - \frac{V_{min}^2}{2}\right)(1 + \lambda V_{DS}) \text{ per } V_{GT} \geq 0$$

$$\text{con } V_{min} = \min(V_{GT}, V_{DS}, V_{DSAT}),$$

$$V_{GT} = V_{GS} - V_T,$$

$$\text{e} \quad V_T = V_{T0} + \gamma(\sqrt{|-2\phi_F + V_{SB}|} - \sqrt{|-2\phi_F|})$$

La Figura 3-24 mostra come il **modello unificato** divida l'intero spazio operativo del transistor in tre regioni (lineare, saturazione di velocità e saturazione) e definisca anche il confine tra queste regioni.

Oltre ad essere funzione delle tensioni applicate ai quattro terminali, il modello utilizza un insieme di cinque parametri: V_{TO}, γ, V_{DSAT}, k', e λ. In linea di principio è possibile ricavare tali parametri dai dati della tecnologia e dalle equazioni della fisica del dispositivo, però la complessità del dispositivo rende questo approccio piuttosto inaffidabile. Un approccio più pratico consiste nell'utilizzare dei valori tali da portare a una buona corrispondenza tra i risultati del

[5] Il modello unificato introduce un termine correttivo per la modulazione della lunghezza di canale anche in regione lineare. Questo fattore, che è piccolo, è solamente funzionale al modello e non ha nessuna corrispondenza nella fisica del dispositivo. Si faccia inoltre attenzione al fatto che l'operatore "min" diviene l'operatore "max" per transistor PMOS.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

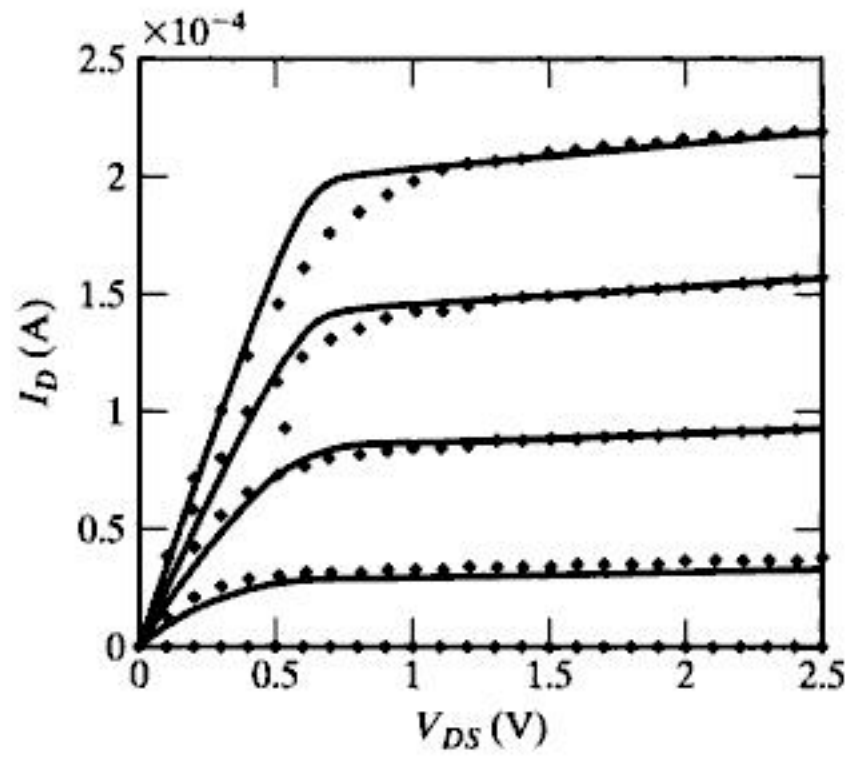


Figura 3-25 Corrispondenza tra il modello semplice (tratto continuo) e la simulazione SPICE (tratteggio) per un transistor NMOS a dimensioni minime (W = 0.375 μm, L = 0.25 μm). Si osservi la discrepanza nella regione di transizione tra comportamento resistivo e saturazione di velocità.

riportate in Figura 3-25 e comparate con i valori simulati per il transistor NMOS. In linea generale vi è una buona corrispondenza, con l'eccezione della zona di transizione tra la regione resistiva e quella di saturazione della velocità. Tale discrepanza, che discende dal modello semplificato della velocità dei portatori dell'Eq. (3.36), è accettabile perché si verifica per i valori più bassi di V_{DS}. Questo esempio dimostra come il modello proposto, sebbene semplificato, riesca a fornire un'indicazione attendibile del comportamento globale del dispositivo.

Parametri per la progettazione—Modello del transistor per l'analisi manuale

La Tabella 3-2 riporta i valori ottenuti per i parametri del modello per un dispositivo NMOS a dimensione minima e per un dispositivo PMOS di dimensioni simili nel nostro processo CMOS generico da 0.25 μm. Questi valori saranno utilizzati nei capitoli seguenti come parametri del modello di un dispositivo generico.

Raccomandazione—Il modello qui riportato è derivato dalle caratteristiche di un singolo dispositivo con lunghezza e larghezza di canale minime. Estrapolare da questo il comportamento di dispositivi con valori di W e L significativamente diversi porta con molta probabilità a errori significativi. Fortunatamente i circuiti digitali utilizzano in genere solo dispositivi a lunghezza minima, poiché questi

Tabella 3-2 Parametri per il modello manuale di un processo CMOS generico da 0.25 μm (dispositivi a lunghezza minima).

	V_{T0} (V)	γ ($V^{0.5}$)	V_{DSAT} (V)	K' (A/V^2)	λ (V^{-1})
NMOS	0.43	0.4	0.63	115×10^{-6}	0.06
PMOS	–0.4	–0.4	–1	-30×10^{-6}	–0.1

Materiale protetto da copyright

portano a minimizzare l'area del sistema. Il grado di corrispondenza tra simulazione e misura per dispositivi di dimensione maggiore sarebbe inaccettabile; è quindi consigliabile estrarre un insieme diverso di parametri da utilizzare per dispositivi con larghezza di canale sensibilmente diversa. ■

Il modello presentato fin qui, basato sull'equivalenza con un *generatore di corrente* si rivelerà molto utile per l'analisi delle proprietà fondamentali di un porta logica, ma a causa della sua non linearità è ancora troppo complesso per analisi più approfondite. Introduciamo quindi un modello ancora più semplificato che è al contempo lineare ed immediato. L'ipotesi di base è che in un sistema digitale il transistor non sia niente altro che un *interruttore* con una resistenza infinita quando è spento ed una resistenza finita R_{on} quando è acceso (vedi Figura 3.26).

Il problema principale di questo modello è che in realtà R_{on} è tempo variante, nonlineare e dipendente dal punto operativo del transistor. Durante l'analisi dei transitori nei circuiti digitali (quando cioè avviene la commutazione tra stati logici diversi) è vantaggioso poter ipotizzare che R_{on} sia una resistenza equivalente R_{eq}, constante e lineare, scelta in modo da far coincidere il risultato del transitorio con quanto si ottiene con il transistor originale. Un approccio ragionevole in questo senso, è quello di scegliere il valor medio della resistenza calcolato nella regione operativa di interesse o, semplificando maggiormente, la media dei due valori della resistenza all'inizio ed alla fine della transizione. Quest'ultimo approccio dà buoni risultati nell'ipotesi che la resistenza non manifesti eccessive non linearità sull'intero intervallo considerato. Di conseguenza otteniamo

$$R_{eq} = \text{media}_{t=t_1\ldots t_2}(R_{on}(t)) = \frac{1}{t_2-t_1}\int_{t_1}^{t_2} R_{on}(t)dt = \frac{1}{t_2-t_1}\int_{t_1}^{t_2}\frac{V_{DS}(t)}{I_D(t)}dt \approx \frac{1}{2}(R_{on}(t_1)+R_{on}(t_2)) \tag{3.41}$$

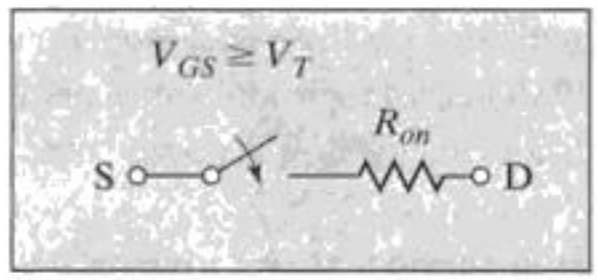
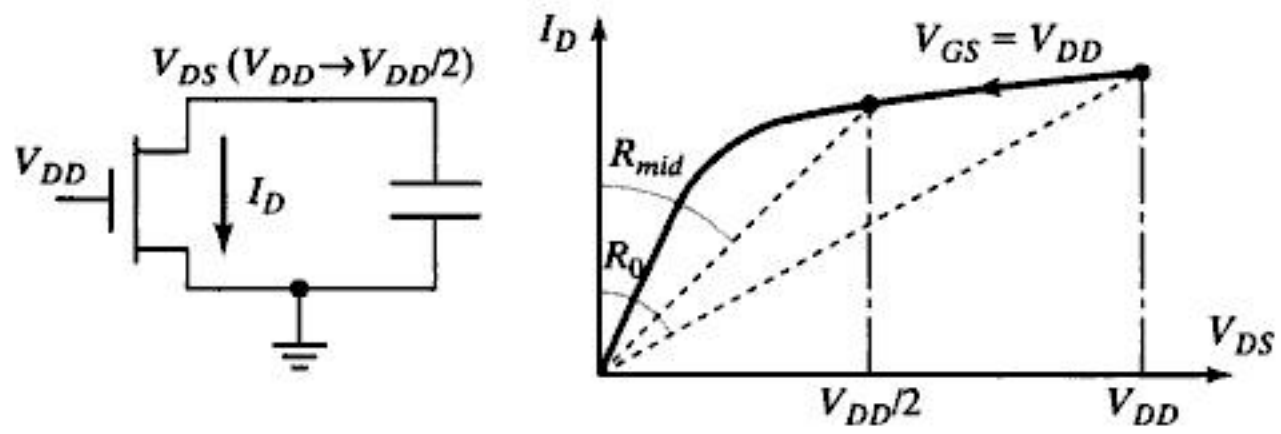


Figura 3-26 Transistor NMOS modellizzato come un interruttore.

Esempio 3.8 Resistenza equivalente durante la carica e scarica di un condesatore

Una delle situazioni più tipiche nei circuiti digitali è la scarica di una capacità da V_{DD} a massa attraverso un transistor NMOS con la tensione di gate pari a V_{DD}, o viceversa—la carica di tale capacità fino a V_{DD} attraverso un PMOS con la tensione di gate pari a GND. Un istante particolarmente significativo è il momento in cui la tensione del condensatore raggiunge $V_{DD}/2$; questo in conseguenza della definizione di tempo di propagazione data nel Capitolo 1. Ipotizzando che la tensione di alimentazione sia sensibilmente più elevata della tensione V_{DSAT} al di sopra della quale avviene la

Materiale protetto da copyright



(a) schematico (b) traiettorie attraversate sul piano ID-VDS

Figura 3-27 Scarica di una capacità attraverso un transistor NMOS: schema circuitale (a) e andamento *I-V* (b). La resistenza istantanea del transistor è uguale a (V_{DS}/I_D) ed è rappresentata dall'angolo rispetto all'asse *y*.

saturazione della velocità dei portatori, è evidente che il transistor opererà in saturazione di velocità per tutta la durata del transitorio. Tale situazione è riportata in Figura 3-27 nel caso di un NMOS che scarica una capacità da V_{DD} a $V_{DD}/2$.

Per mezzo dell'Eq. (3.41), possiamo ricavare il valore della resistenza equivalente, che corrisponde al valore medio della resistenza del dispositivo su tutto l'intervallo di commutazione:

$$R_{eq} = \frac{1}{-V_{DD}/2} \int_{V_{DD}}^{V_{DD}/2} \frac{V}{I_{DSAT}(1+\lambda V)} dV \approx \frac{3}{4}\frac{V_{DD}}{I_{DSAT}}\left(1 - \frac{7}{9}\lambda V_{DD}\right)$$
$$\text{con} \quad I_{DSAT} = k'\frac{W}{L}\left((V_{DD} - V_T)V_{DSAT} - \frac{V_{DSAT}^2}{2}\right) \tag{3.42}$$

Un risultato del tutto simile si ottiene prendendo semplicemente la media tra i due valori della resistenza ai due estremi della commutazione (e semplificando il risultato mediante uno sviluppo in serie di Taylor):

$$R_{eq} = \frac{1}{2}\left(\frac{V_{DD}}{I_{DSAT}(1+\lambda V_{DD})} + \frac{V_{DD}/2}{I_{DSAT}(1+\lambda V_{DD}/2)}\right) \approx \frac{3}{4}\frac{V_{DD}}{I_{DSAT}}\left(1 - \frac{5}{6}\lambda V_{DD}\right) \tag{3.43}$$

Da queste espressioni possiamo dedurre tre conclusioni significative:

- La resistenza è inversamente proporzionale al rapporto (*W/L*) del dispositivo. Raddoppiare la larghezza del transistor significa ridurre a metà la resistenza.
- Per $V_{DD} >> V_T + V_{DSAT}/2$ la resistenza diviene praticamente indipendente dalla tensione di alimentazione. Questo fatto è confermato dalla Figura 3-28, che riporta la resistenza equivalente (simulata) in funzione della tensione di alimentazione V_{DD}. In corrispondenza dell'aumento della tensione di alimentazione si

Materiale protetto da copyright

nota un miglioramento molto ridotto in termini di resistenza, essenzialmente dovuto alla modulazione della lunghezza del canale.

- Una volta che la tensione di alimentazione raggiunge V_T, la resistenza aumenta in modo drastico.

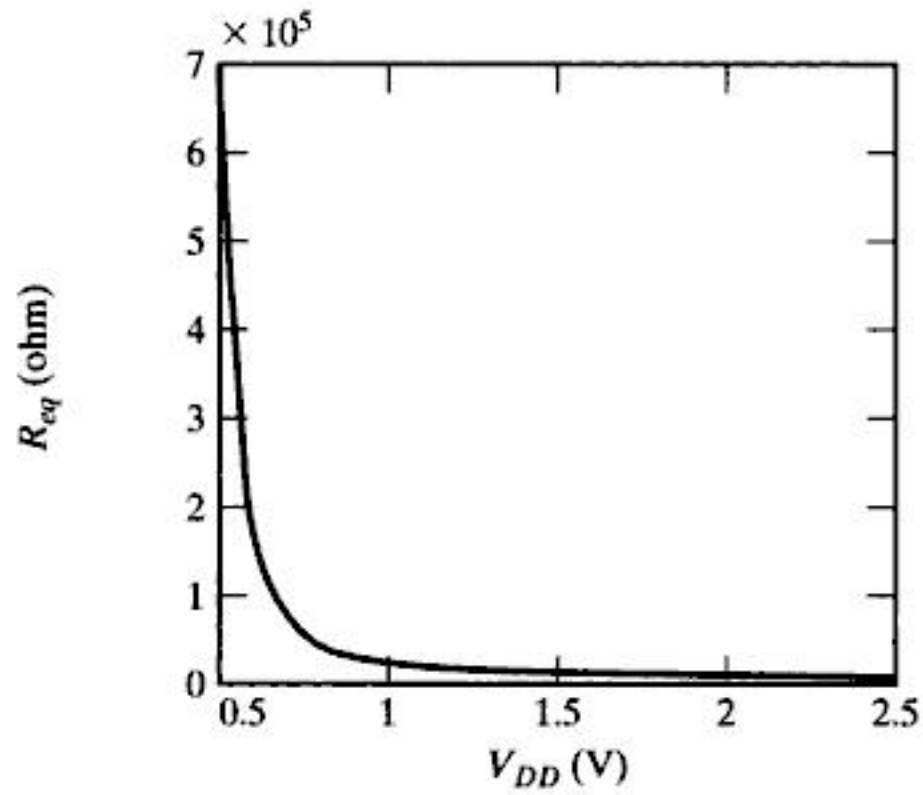


Figura 3-28 Simulazione della resistenza equivalente per un transistor NMOS a dimensione minima in un processo CMOS da 0.25 µm in funzione di V_{DD} ($V_{GS} = V_{DD}$, $V_{DS} = V_{DD} \rightarrow V_{DD}/2$).

Parametri per la progettazione — Modello della resistenza equivalente

La Tabella 3-3 elenca i valori della resistenza equivalente ottenuti tramite simulazione per il nostro processo CMOS generico da 0.25 µm. Questi valori saranno utili per analizzare le prestazioni delle porte logiche CMOS discusse nei prossimi capitoli.

Tabella 3-3 Resistenza equivalente R_{eq} (W/L = 1) di transistor NMOS e PMOS in un processo CMOS da 0.25 µm (con $L = L_{min}$). Per dispositivi più larghi, dividere R_{eq} per W/L.

V_{DD} (V)	1	1.5	2	2.5
NMOS (kΩ)	35	19	15	13
PMOS (kΩ)	115	55	38	31

■

3.3.3 Il comportamento dinamico del transistor MOS

La risposta dinamica di un transistor MOSFET è esclusivamente funzione del tempo necessario a caricare o scaricare le capacità parassite che sono intrinseche al dispositivo e le capacità aggiuntive introdotte dal carico e dalle linee di interconnessione (questo argomento sarà trattato nel Capitolo 4). Una comprensione approfondita della natura e del comportamento di tali

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

stor è spento, non sussiste nessun canale e la capacità totale, uguale a WLC_{ox}, appare tra gate e substrato. Al crescere di V_{GS}, la regione di svuotamento si forma sotto al gate. Questo effetto è equivalente ad un aumento dello spessore del dielettrico di gate e quindi comporta una diminuzione della capacità. Una volta che il transistor è accesso ($V_{GS} = V_T$), il canale è formato e C_{GCB} si riduce a 0. Con $V_{DS} = 0$, il dispositivo opera in regione resistiva e la capacità si divide equamente tra source e drain ($C_{GCS} = C_{GCD} = WLC_{ox}/2$). La notevole variazione della capacità di gate intorno al punto in cui $V_{GS} = V_T$ va tenuta in debita considerazione: un progettista che necessiti di una capacità molto lineare dovrebbe evitare tale regione operativa.

Una volta che il transistor è accesso, la distribuzione della capacità di gate dipende fortemente dal grado di saturazione misurato dal rapporto $V_{DS}/(V_{GS} - V_T)$. Come mostrato in Figura 3-31b, C_{GCD} scende gradualmente a zero in corrispondenza dell'aumento del livello di saturazione, mentre C_{GCS} cresce fino a 2/3 $C_{ox}WL$. Questo significa anche che la capacità totale di gate diminuisce per un livello maggiore di saturazione.

Questa trattazione ha messo in evidenza come le componenti della capacità di gate siano non lineari e varino a seconda delle tensioni di polarizzazione. Al fine di svolgere un'analisi del primo ordine, adotteremo un modello semplificato lineare a tratti, che assume un valore costante della capacità per ciascuna regione operativa. I valori utilizzati sono riassunti nella Tabella 3-4.

Tabella 3-4 Valore medio della capacità di canale del transistor MOS per diverse regioni operative.

Regione operativa	C_{GCB}	C_{GCS}	C_{GCD}	C_{GC}	C_G
Spento	$C_{ox}WL$	0	0	$C_{ox}WL$	$C_{ox}WL + 2C_oW$
Resistiva	0	$C_{ox}WL/2$	$C_{ox}WL/2$	$C_{ox}WL$	$C_{ox}WL + 2C_oW$
Saturazione	0	$(2/3)C_{ox}WL$	0	$(2/3)C_{ox}WL$	$(2/3)C_{ox}WL + 2C_oW$

Esempio 3.9 Utilizzare un simulatore circuitale per estrarre la capacità

Determinare il valore delle capacità parassite di un transistor MOS per una data condizione operativa è un lavoro faticoso e richiede la conoscenza di una serie di parametri della tecnologia che spesso non sono esplicitamente disponibili. Fortunatamente, una volta che un modello SPICE del transistor è disponibile, una semplice simulazione può fornire tutti i dati necessari. Ipotizziamo di voler conoscere il valore della capacità di gate totale in funzione di V_{GS} (per $V_{DS} = 0$). Una simulazione del circuito di Figura 3-32a ci fornirà esattamente tale informazione. Di fatto la relazione

$$I = C_G(V_{GS})\frac{dV_{GS}}{dt}$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

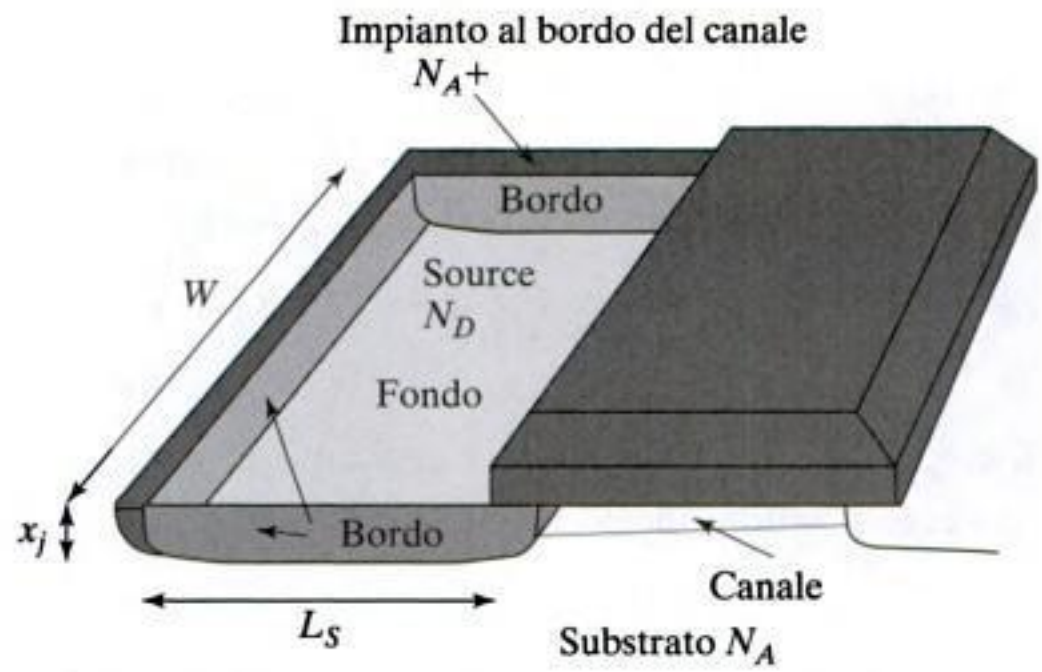


Figura 3-33 Vista dettagliata della giunzione di source.

zione è tipicamente di tipo brusco, il coefficiente di profilo della giunzione m raggiunge il valore 0.5.

- Il *bordo* della giunzione (*side-wall*), formata dalla regione di source con drogaggio N_D e l'impianto p^+ al bordo del canale (*channel-stop implant*) con livello di drogaggio N_A^+. Il livello di drogaggio dello stopper è maggiore di quello del substrato e ne consegue che la capacità per unità di area è maggiore. La giunzione di bordo tipicamente è graduale e il coefficiente di profilo della giunzione varia da 0.33 a 0.5. Il valore della capacità è dato da $C_{sw} = C'_{sw}x_j(W + 2 \times L_s)$. Si noti che non viene contato alcun contributo di capacità di bordo per il quarto lato dell'area di source, poiché questo coincide con il canale conduttivo[7]. Dato che la profondità della giunzione x_j è un parametro della tecnologia, viene normalmente combinato con C'_{jsw} ottenendo una capacità per unità di perimetro $C_{jsw} = C'_{jsw}x_j$. A questo punto possiamo ricavare un'espressione per la capacità di giunzione totale come segue:

$$\begin{aligned} C_{diff} &= C_{bottom} + C_{sw} = C_j \times AREA + C_{jsw} \times PERIMETRO \\ &= C_j L_S W + C_{jsw}(2L_S + W) \end{aligned} \tag{3.45}$$

Dato che tutte queste sono capacità di piccolo segnale, generalmente vengono linearizzate e se ne utilizza un valore medio simile a quello definito nell'Eq. (3.10).

Problema 3.1 Utilizzo di un simulatore circuitale per determinare la capacità di drain

Utilizzando un simulatore circuitale (in analogia alla Figure 3.32), derivare un circuito semplice che sia di aiuto per determinare la capacità di drain di un transistor NMOS nelle differenti modalità operative.

[7]Per correttezza dovremmo tener conto anche della capacità di diffusione della giunzione source(drain)-canale. Considerate le condizioni di drogaggio e l'area limitata, tale componente può essere quasi del tutto ignorata in un'analisi del primo ordine. I modelli dettagliati di SPICE aggiungono un fattore C_{JSWG} per tener conto di tale giunzione.

Materiale protetto da copyright

Modello delle capacità del dispositivo

Tutti i contributi discussi in precedenza possono essere combinati in un unico modello per le capacità del transistor MOS, che è mostrato in Figura 3-34. I vari componenti discussi in precedenza possono essere riconosciuti con facilità. L'equazione è

$$C_{GS} = C_{GCS} + C_{GSO};\ C_{GD} = C_{GCD} + C_{GDO};\ C_{GB} = C_{GCB}$$
$$C_{SB} = C_{Sdiff};\ C_{DB} = C_{Ddiff} \tag{3.46}$$

Per il progettista di circuiti ad alte prestazioni e consumi di potenza ridotti è di fondamentale importanza conoscere a fondo questo modello ed avere familiarità con i valori di ciascun componente.

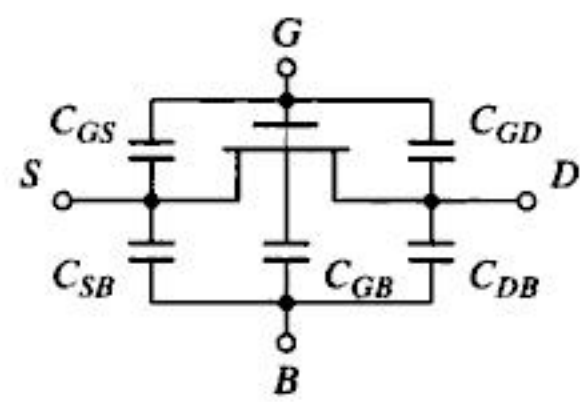


Figura 3-34 Modello delle capacità del MOSFET.

Esempio 3.10 Capacità del transistor MOS

Consideriamo un transistor NMOS con i seguenti parametri: t_{ox} = 6 nm, L = 0.24 μm, W = 0.36 μm, $L_D = L_S$ = 0.625 μm, $C_O = 3 \times 10^{-10}$ F/m, $C_{j0} = 2 \times 10^{-3}$ F/m^2, e $C_{jsw0} = 2.75 \times 10^{-10}$ F/m. Determinare il valore di tutti i contributi capacitivi rilevanti in condizione di polarizzazione nulla.

La capacità di gate per unità di area si calcola facilmente come $\varepsilon_{ox} / t_{ox}$ e vale 5.7 fF/μm^2. La capacità gate-canale C_{GC} è quindi WLC_{ox} = 0.49 fF. Per trovare la capacità di gate totale dobbiamo aggiungere i contributi di sovrapposizione gate-source e gate-drain, ognuno dei quali vale WC_O = 0.105 fF. Si arriva quindi ad un valore totale per la capacità di gate pari a 0.7 fF.

La capacità di diffusione è costituita dal contributo della base e dei bordi dell'area attiva. Il primo contributo vale $C_{j0}\ L_D W$ = 0.45 fF, mentre la capacità del bordo per polarizzazione nulla vale $C_{jsw0}\ (2L_D + W)$ = 0.44 fF. Di conseguenza si ottiene una valore totale per la capacità di drain (source)-bulk uguale a 0.89 fF.

La capacità di diffusione pare dominante rispetto a quella di gate. Si tratta però di una configurazione al caso peggiore. Quando si aumenta il valore della tensione inversa ai capi della giunzione, come accade durante il funzionamento normale dei circuiti MOS, la capacità di diffusione si riduce sensibilmente. Inoltre una progettazione ottimizzata può aiutare a ridurre il valore di L_D (L_S). In generale possiamo dire che le capacità di diffusione sono al più uguali, e molto spesso minori, della capacità di gate.

Materiale protetto da copyright

Parametri per la progettazione—Capacità del Transistor MOS

La Tabella 3-5 riporta i parametri necessari per quantificare le capacità parassite del transistor MOS nel nostro processo CMOS da 0.25 μm.

Tabella 3-5 Parametri capacitivi per i transistor NMOS e PMOS nel processo CMOS da 0.25μm.

	C_{ox} (fF/μm²)	C_O (fF/μm)	C_j (fF/μm²)	m_j	ϕ_b (V)	C_{jsw} (fF/μm)	m_{jsw}	ϕ_{bsw} (V)
NMOS	6	0.31	2	0.5	0.9	0.28	0.44	0.9
PMOS	6	0.27	1.9	0.48	0.9	0.22	0.32	0.9

■

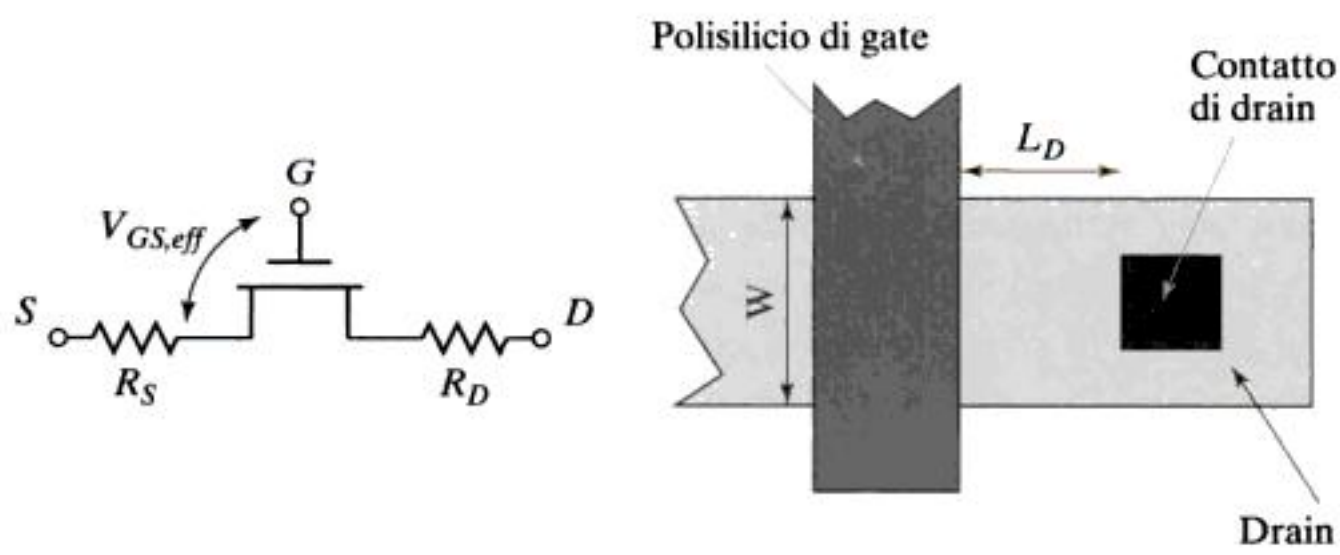
Resistenza source–drain

Le prestazioni di un circuito CMOS possono essere compromesse anche da altri parametri parassiti, in particolare le resistenze parassite delle diffusioni di drain e di source, come mostrato in Figura 3-35a. Questo effetto è più evidente al ridursi delle dimensioni minime del transistor, poiché le giunzioni sono meno profonde e l'area di contatto si restringe. La resistenza della regione di drain (source) può essere espressa come

$$R_{S,D} = \frac{L_{S,D}}{W} R_{\square} + R_C, \tag{3.47}$$

dove R_C è la resistenza di contatto, W è la larghezza del transistor e $L_{S,D}$ è la lunghezza della regione di source o drain (Figura 3-35b). $R_{\square}$, detta *sheet resistance*, è la resistenza di una diffusione quadrata di drain o source ed è compresa tra 20 e 100 Ω/□. Si osservi che la resistenza di un quadrato di un dato materiale è costante, indipendentemente dalla sue dimensioni.

La resistenza serie causa una degradazione delle prestazioni del dispositivo, perché ne riduce la corrente di drain per una data tensione di controllo. Contenere il più possibile il suo valore è quindi un obiettivo primario sia per il progettista che per l'ingegnere di processo.



(a) Modello della resistenza serie (b) Parametri della resistenza serie

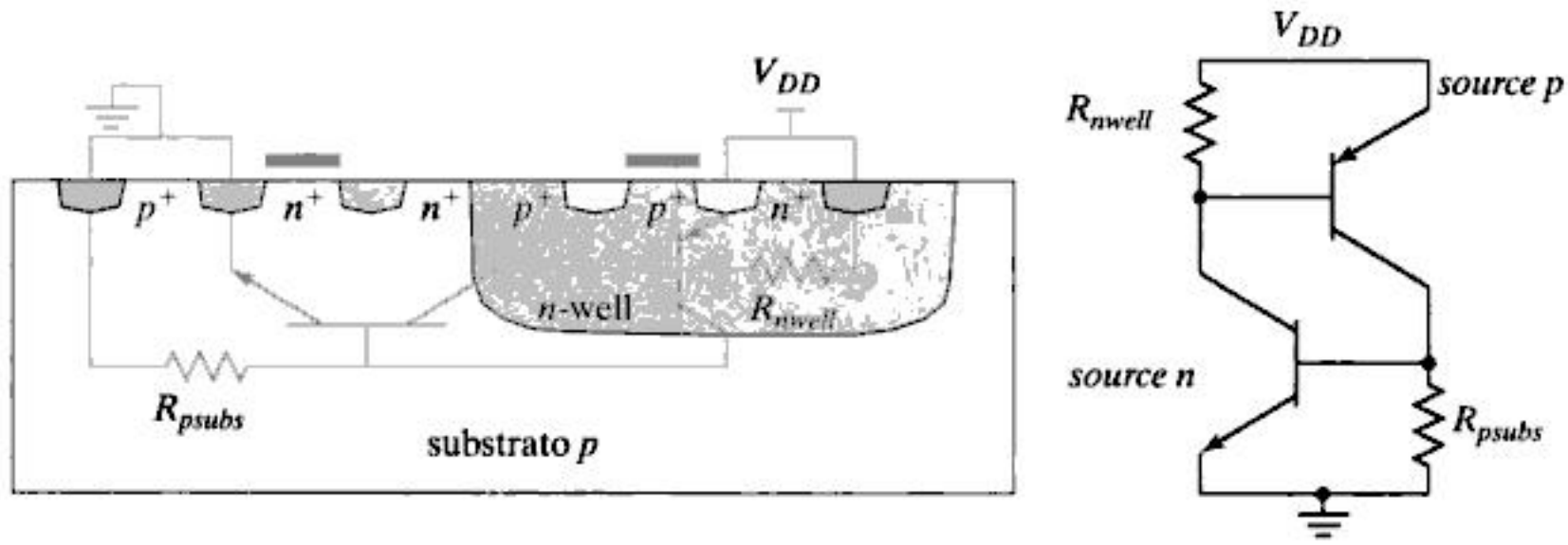
Figura 3-35 Resistenze parassite di drain e di source.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Origine del latchup (b) Circuito equivalente

Figura 3-38 Latchup CMOS.

un circuito equivalente è riportato in Figura 3-38b. Quando uno dei due transistor bipolari è polarizzato in zona attiva diretta (ad esempio a causa della corrente che scorre nella well o nel substrato), fornisce corrente alla base dell'altro transistor. Tale retroazione positiva fa aumentare la corrente finché il circuito smette di funzionare correttamente o addirittura si brucia.

Da questa analisi discende un messaggio chiaro per il progettista: per evitare il latchup, le resistenze R_{nwell} e R_{psub} devono essere rese minime. Questo si può fare inserendo numerosi contatti di well e substrato, piazzati vicino alle connessioni di source dei dispositivi NMOS e PMOS. I dispositivi che devono sostenere correnti elevate (come i transistor dei driver di I/O) dovrebbero essere circondati da anelli di guardia (*guard ring*). Tali contatti di well/substrato, posti attorno al transistor, riducono ulteriormente la resistenza e minimizzano il guadagno del bipolare parassita. Il latchup era realmente critico nei primi processi CMOS. Più recentemente le innovazioni del processo e le migliori tecniche progettuali hanno eliminato quasi del tutto il rischio del latchup.

3.3.5 Modelli SPICE per il transistor MOS

La complessità del comportamento del transistor a canale corto e i suoi numerosi parametri parassiti hanno portato allo sviluppo di una serie di modelli con diversi gradi di accuratezza ed efficienza computazionale. In linea generale maggior accuratezza comporta maggiore complessità e quindi maggiori tempi di calcolo. In questo paragrafo descriviamo brevemente le caratteristiche dei modelli più popolari del transistor MOS e come istanziare un transistor nella descrizione circuitale per la simulazione.

Modelli SPICE

SPICE possiede tre diversi modelli MOS predefiniti selezionabili attraverso il parametro LEVEL nella dichiarazione del modello. Sfortunatamente tutti questi modelli sono ormai

[9]Questa discussione sui meccanismi che causano il latchup è mantenuta ad alto livello di astrazione per necessità. I lettori interessati possono trovare discussioni approfondite nei libri che trattano di dispositivi, come [Streetman95].

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

3.4 Una parola sulle variazioni del processo

Nella trattazione fatta fin qui, abbiamo assunto che un dispositivo possa essere descritto adeguatamente da un singolo insieme di parametri. In realtà i parametri di un transistor variano da wafer a wafer, o addirittura da quelli di altri transistor sullo stesso chip, a seconda della posizione. Questa distribuzione casuale che si osserva tra dispositivi che dovrebbero in teoria essere identici, è dovuta principalmente a due fattori:

1. Variazioni dei parametri di processo, quali densità della concentrazione di impurità, spessore dell'ossido e profondità della diffusioni, causate da condizioni non uniformi durante la deposizione e/o diffusione delle impurità. Queste comportano valori alterati per le resistenze di quadro e per parametri del transistor quale la tensione di soglia.
2. Variazioni delle dimensioni dei dispositivi, principalmente dovute alla risoluzione limitata del processo fotolitografico. Questo comporta alterazioni del rapporto (W/L) dei transistor MOS e della larghezza delle piste di interconnessione.

Osserviamo che un numero sostanziale di queste deviazioni sono completamente incorrelate tra loro. Per esempio le variazioni della lunghezza di canale di un transistor MOS sono incorrelate con le variazioni della sua tensione di soglia, poiché entrambe sono definiti da passi differenti del processo.

- Queste variazioni di processo influenzano i parametri che determinano le prestazioni di un circuito. Si pensi, per esempio, alla corrente nel transistor e ai suoi parametri.
- La *tensione di soglia* V_T può variare per differenti motivi: cambiamenti nello spessore dell'ossido, livelli di impurità nel substrato, nel polisilicio e negli impianti e carica superficiale. Un controllo preciso della tensione di soglia è un fattore molto importante. Mentre in passato la tensione di soglia poteva variare fino al 50%, nei processi digitali allo stato dell'arte si è in grado di controllare la soglia entro 25–50 mV.
- La causa principale delle variazioni della transconduttanza del processo k'_n è la variazione dello spessore dell'ossido. Possono anche verificarsi delle variazioni della mobilità, ma in grado minore.
- Variazioni in W e L. Queste sono causate principalmente dal processo litografico. Si noti che le variazioni di W e L sono totalmente incorrelate tra loro perché la larghezza è determinata dall'area attiva e dalla deposizione dell'ossido di campo, mentre la lunghezza è definita dalla deposizione del polisilicio e dalla diffusione delle regioni di source e drain.

L'effetto tangibile delle variazioni di processo può essere una deviazione sostanziale del comportamento del circuito dalla risposta nominale o attesa, sia in senso positivo che negativo. Questo fatto pone il progettista di fronte ad una scelta di carattere economico. Assumiamo, per esempio, di dover progettare un microprocessore che lavori con una frequenza di clock pari a 3 GHz. Dal punto di vista economico, è importante che la maggior parte dei chip costruiti rispetti tale richiesta in termini di prestazioni. Un modo per raggiungere tale obiettivo potrebbe essere quello di assumere il valore del caso peggiore per tutti i

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Tabella 3-8 Tipi di scaling tecnologico per dispositivi a canale corto.

Parametri	Espressioni	Scaling a campo costante	Scaling generale	Scaling a tensione costante
W, L, t_{ox}		$1/S$	$1/S$	$1/S$
V_{DD}, V_T		$1/S$	$1/U$	1
N_{SUB}	V/W_{depl}^2	S	S^2/U	S^2
Area/Dispositivo	WL	$1/S^2$	$1/S^2$	$1/S^2$
C_{ox}	$1/t_{ox}$	S	S	S
C_{gate}	$C_{ox}WL$	$1/S$	$1/S$	$1/S$
k_n, k_p	$C_{ox}W/L$	S	S	S
I_{sat}	$C_{ox}WV$	$1/S$	$1/U$	1
Densità di corrente	$I_{sat}/Area$	S	S^2/U	S^2
R_{on}	V/I_{sat}	1	1	1
Ritardo intrinseco	$R_{on}C_{gate}$	$1/S$	$1/S$	$1/S$
P	$I_{sat}V$	$1/S^2$	$1/U^2$	1
Densità di potenza	$P/Area$	1	S^2/U^2	S^2

della corrente di saturazione dalla tensione di controllo (in accordo all'Eq. (3.38)). Discutiamo quindi ciascuno dei possibili scenari.

Scaling a campo costante (scaling completo)

In questo modello dimensioni e tensioni sono scalate dello stesso fattore S. L'obiettivo principale in questo caso è di mantenere costanti le intensità dei campi elettrici nei dispositivi scalati, rispetto ai dispositivi originali. Mantenere i campi elettrici costanti garantisce l'integrità dei dispositivi ed evita fenomeni di breakdown o altri effetti secondari. Questo tipo di evoluzione delle geometrie porta a maggior densità di integrazione (*area*), maggiori prestazioni (*ritardo intrinseco*), e minor consumo di potenza (P). Gli effetti di uno scaling completo sui dispositivi e sui parametri del circuito sono elencati nella terza colonna della Tabella 3-8. Utilizziamo la costante di tempo intrinseca, cioè il prodotto tra la capacità di gate e la resistenza

Materiale protetto da copyright

equivalente del transistor acesso, come una figura di merito. L'analisi mostra che la resistenza equivalente rimane costante a causa della riduzione contemporanea dell'escursione logica e della corrente. Il miglioramento in termini di prestazioni è quindi dovuto solamente alla diminuzione delle capacità. Il risultato ottenuto evidenzia gli effetti benefici del riscalamento: la velocità del circuito cresce linearmente mentre la potenza dissipata dalla porta logica diminuisce con andamento quadratico.[10]

Scaling a tensione costante

Nella realtà lo scaling completo non è praticabile: per mantenere i nuovi dispositivi compatibili con quelli già esistenti, le tensioni non possono essere scalate in modo arbitrario; infatti prevedere più tensioni di alimentazione in un sistema comporta notevoli costi aggiuntivi. Di conseguenza le tensioni in gioco non vengono scalate quanto le dimensioni minime e i progettisti rispettano standard ben definiti per le tensioni di alimentazione e per i livelli del segnale. Come mostrato in Figura 3-41, 5 V era lo *standard de facto* per tutti i componenti digitali fino ai primi anni '90: si utilizzò quindi fino ad allora un modello di *scaling a tensione costante*.

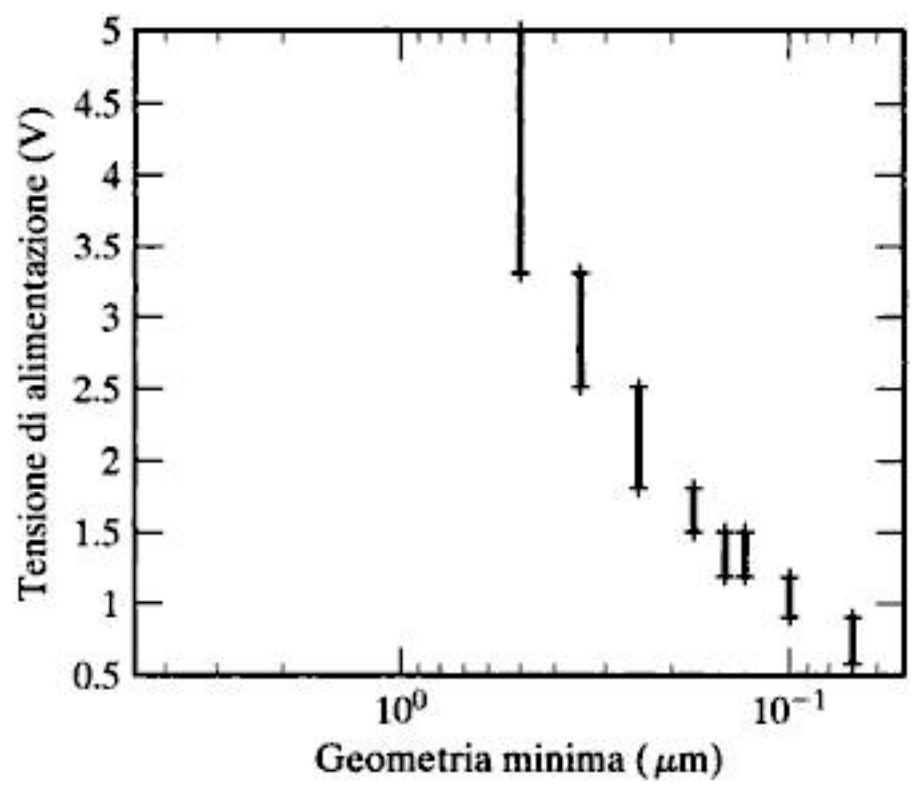


Figura 3-41 Evoluzione della tensione di alimentazione minima e massima in circuiti integrati digitali, in funzione della geometria minima.

[10]La compilazione della Tabella 3-8 è anche frutto di alcune ipotesi.

1. Si è ipotizzato che la mobilità dei portatori non sia influenzata dalla riduzione delle geometrie.
2. Il drogaggio del substrato N_{sub} è stato scalato in modo che la larghezza massima della regione di svuotamento fosse ridotta di un fattore S.
3. Si è anche ipotizzato che il ritardo introdotto da un dispositivo sia dovuto principalmente alla capacità intrinseca (la capacità di gate), e che le altre capacità dei dispositivi (quali le capacità di diffusione) scalino opportunamente.Questa ipotesi è approssimativamente vera nel caso dello scaling completo, ma non lo è per lo scaling a tensione costante, dove C_{diff} scala come $1/\sqrt{S}$.

Materiale protetto da copyright

I nuovi standard a 3.3 V e 2.5 V cominciarono ad essere utilizzati solo con l'avvento delle tecnologie CMOS da 0.5 μm. Oggigiorno si comincia a vedere una più stretta relazione tra tensioni e dimensioni dei dispositivi. Il motivo di questo cambiamento nel modello di scaling si comprende in parte analizzando il modello a tensione costante, evidenziato nella quinta colonna della Tabella 3-8. In un dispositivo in saturazione di velocità, mantenere costanti le tensioni mentre si riducono le dimensioni dei dispositivi, non porta nessun miglioramento in termini di prestazioni rispetto a un modello di scaling completo, ma anzi comporta un aggravio in termini di potenza dissipata. Il guadagno in termini di maggior corrente disponibile è semplicemente annullato dal livello di tensione maggiore, il che ha come unico risultato l'aumento della potenza dissipata. Questa situazione è molto diversa da quella che si presentava con i dispositivi a canale lungo, nei quali la corrente dipende in modo quadratico dalla tensione di gate (si ricordi l'Eq. (3.29)). Mantenere costanti le tensioni in quella situazione porta un vantaggio netto in termini di prestazioni, perché si ha una riduzione effettiva della resistenza equivalente. Altre ragioni a sostegno della riduzione della tensione di alimentazione sono legate a fenomeni fisici quali l'effetto dei portatori caldi e il breakdown dell'ossido e rendono effettivamente impraticabile lo scaling a tensione costante.

Problema 3.2 Riduzione delle geometrie nei dispositivi a canale lungo

Dimostrare che, per un transistor a canale lungo, lo scaling completo implica una riduzione del ritardo intrinseco di un fattore S^2, mentre la potenza dissipata cresce di un fattore S. Ricostruire la Tabella 3-8, ipotizzando che la corrente abbia una dipendenza quadratica rispetto alla tensione (Eq. (3.29)).

NOTA: La situazione descritta nel paragrafo precedente è solo un modello al primo ordine. L'aumento della tensione di alimentazione in transistor a canale corto comporta comunque alcuni vantaggi in termini di prestazioni. Questo è evidente nella Figura 3-28 e nella Tabella 3-3, che evidenziano una diminuzione della resistenza equivalente in corrispondenza dell'aumento della tensione di alimentazione persino in regime di alta tensione. Ciò nonostante, questo effetto, che è principalmente imputabile alla modulazione della lunghezza del canale, è secondario e molto minore di quanto si otterrebbe con un dispositivo a canale lungo.

Il lettore dovrebbe tener presente questo avvertimento nello studio dell'evoluzione delle tecnologie. L'obiettivo principale dell'analisi è però quello di identificare gli andamenti al primo ordine della varie figure di merito, ignorando quindi ogni effetto del secondo ordine, quali degradazione della mobilità, resistenze serie, ecc.

Scaling generale

Possiamo osservare dalla Figura 3-41 che le tensioni di alimentazione, sebbene diminuiscano, non scalano con la stessa velocità della tecnologia. Per esempio, passando da una tecnologia da 0.5 μm ad una da 0.1 μm, la massima tensione di alimentazione da 5 V si riduce solo fino a 1.5 V. La domanda ovvia a questo punto è: perché non utilizzare sempre il modello di scaling a campo costante, dato che mantenere la tensione di alimentazione a un

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

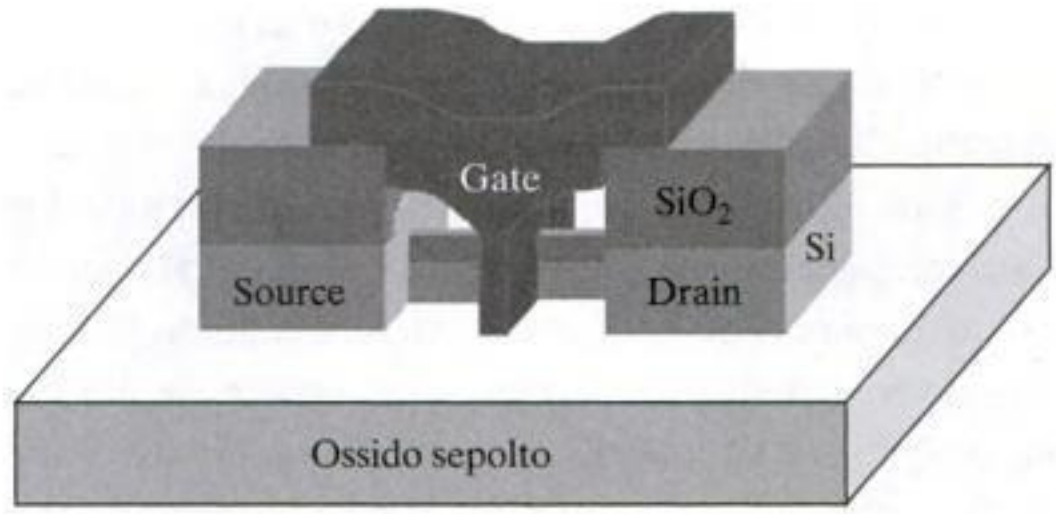


Figura 3-42 Transistor FinFET a doppio gate con lunghezza di canale da 25 nm [Huang99].

tinuerà questa tendenza, certo è che sono molte le discussioni in proposito. È già stato dimostrato in laboratorio che dispositivi CMOS con lunghezza di canale al di sotto dei 20-nm funzionano correttamente ed hanno caratteristiche di corrente incredibilmente simili a quelle dei transistor odierni. Questi transistor, sebbene operino sulla base degli stessi concetti degli attuali dispositivi MOS, sono sostanzialmente diversi dalle strutture con cui siamo familiari e richiedono una sostanziale ottimizzazione del dispositivo. La Figura 3-42, per esempio, mostra una possibile struttura di transistor, il transistor FinFET dual-gated di Berkeley, che si è dimostrato funzionale fino a lunghezze di canale estremamente ridotte.

Un'altra possibilità è il *transistor verticale*. Sebbene l'introduzione di molti livelli di metallo abbia trasformato i circuiti integrati in strutture tridimensionali, il layout del transistor è in effetti fondamentalmente sviluppato su due dimensioni. Questo costringe i progettisti dei dispositivi a ottimizzare contemporaneamente la densità di integrazione e le prestazioni; ruotando il dispositivo in modo che il drain sia in cima ed il source sul fondo, le due problematiche sono separate: la densità di integrazione è ancora dominata dalla dimensione orizzontale, mentre le prestazioni dipendono dalle geometrie verticali (vedi Figura 3-43). Dispositivi correttamente funzionanti di questo tipo sono stati fabbricati con lunghezze di canale sostanzialmente al di sotto di 0.1 μm [Eaglesham00].

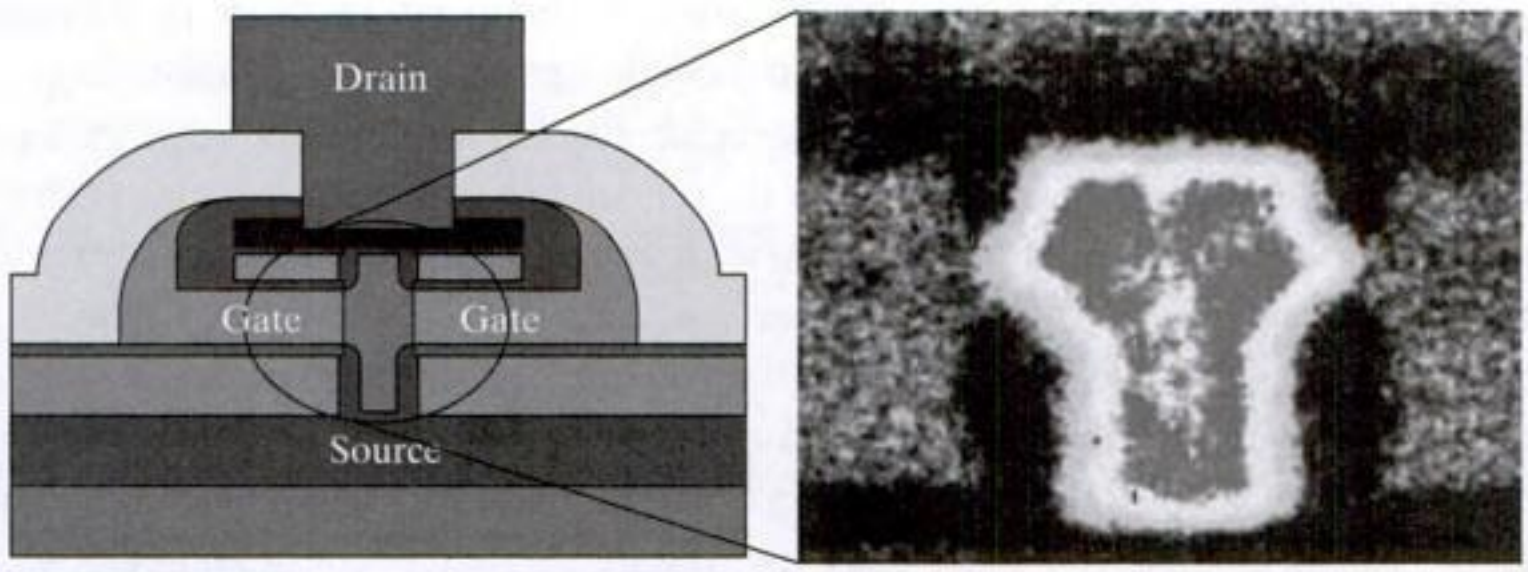


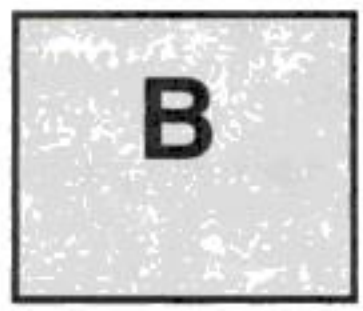
Figura 3-43 Transistor verticale a doppio gate. La fotografia a destra mostra una vista ingrandita della regione di canale.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



Simulazione circuitale

Proprietà della simulazione circuitale

Modelli per la simulazione circuitale

L'aspettativa principale di un progettista rispetto all'automazione delle varie fasi di sintesi è la disponibilità di strumenti di analisi accurati e veloci. Il primo strumento di progettazione assistita al calcolatore (*computer-aided design, CAD*) ad essere accettato estesamente fu il simulatore circuitale SPICE, che senza dubbio è l'ausilio computerizzato per la progettazione di circuiti (digitali) più utilizzato in assoluto. Inizialmente SPICE fu sviluppato presso la University of California a Berkeley [Nagel75], dove vennero rilasciate tre versioni principali (Spice-1, 2 e 3) nel corso degli anni. Successivamente sono stati introdotti diversi simulatori derivati da SPICE (quali PSPICE o HSPICE), estendendo a livello commerciale uno strumento originariamente concepito in una università. Negli ultimi anni abbiamo poi assistito all'avvento di un certo numero di simulatori circuitali (quali SPECTRE [Spectre], e ELDO [Eldo]), che hanno introdotto prestazioni o precisioni migliori, grazie a nuovi algoritmi e a tecniche di programmazione moderne.

In questo testo utilizziamo estesamente la simulazione circuitale per illustrare i concetti fondamentali dei circuiti digitali e per validare le nostre analisi manuali. Anche gli esercizi del libro si affidano pesantemente alla simulazione circuitale, quindi è necessario acquisire una buona familiarità con le possibilità e le peculiarità della simulazione e dell'analisi a questo livello di astrazione.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

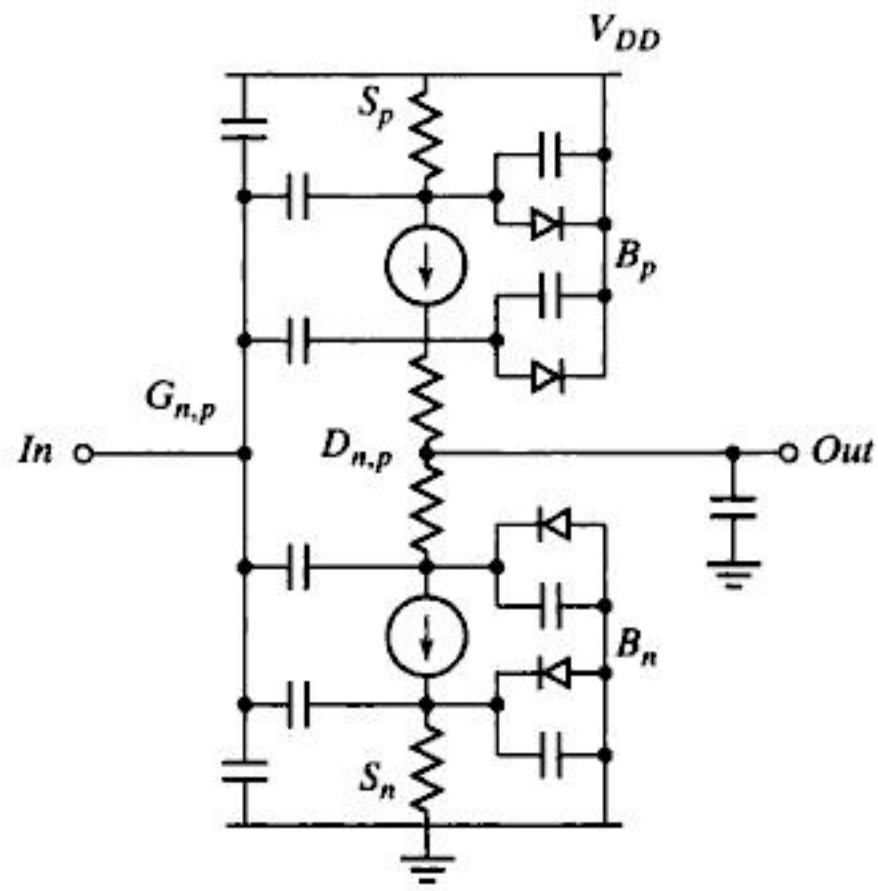


Figura B-1 Modello dell'invertitore CMOS per la simulazione circuitale.

Analisi al transitorio Questa è l'analisi utilizzata con maggiore frequenza nella simulazione dei circuiti digitali. L'analisi al transitorio calcola l'operazione del circuito mentre il tempo progredisce. Questo è lo strumento ideale per determinare i parametri caratteristici dei transitori, quali tempi di propagazione, salita e discesa e dissipazione di potenza. L'analisi è controllata con la linea seguente:

.TRAN 1ps 250ps

Il comando specifica l'intervallo di simulazione (250 ps), così come la frequenza con cui memorizzare i dati (ogni ps). Quest'ultimo parametro non è direttamente correlato con il passo di simulazione effettivamente utilizzato; il simulatore stesso fissa il passo ottimale basandosi sull'attività del circuito.

SPICE (o un simulatore equivalente) fornisce un insieme completo di strumenti aggiuntivi per aiutare il progettista ad ottenere tutte le risposte necessarie in modo efficiente. La pagina web del libro mette a disposizione un vasto insieme di utili suggerimenti ed informazioni per il lettore interessato.

Modelli dei dispositivi

Mentre la maggior parte dei simulatori circuitali utilizzano tecniche di soluzione numerica molto simili tra loro, ciò che li differenzia sostanzialmente sono la robustezza, l'efficienza e la precisione dei modelli dei dispositivi. Per esempio, è possibile definire modelli molto elaborati e precisi che però risultano completamente inutili per problemi di convergenza degli algoritmi numerici. La complessità del comportamento dei transistor MOS a canale corto ed i relativi molti effetti parassiti hanno portato a numerosi modelli con vari livelli di precisione ed efficienza computazionale, alcuni dei quali sono stati presentati nel capitolo precedente.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

[Nagel75] L. Nagel, "SPICE2: a Computer Program to Simulate Semiconductor Circuits," Memo ERL-M520, Dept. Elect. and Computer Science, University of California at Berkeley, 1975.

[Rabaey96] J. Rabaey (et al), "The Spice Page," *http://bwrc.eecs.berkeley.edu/ICbook/spice*, 1996.

[Spectre] "Spectre Circuit Simulator," *http://www.cadence.com/datasheets/spectre_cir_sim.html*, Cadence.

[Thorpe92] T. Thorpe, *Computerized Circuit Analysis with SPICE*, John Wiley and Sons, 1992.

[Vladimirescu93] A. Vladimirescu, *The SPICE Book*, John Wiley and Sons, 1993.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

CAPITOLO

Le connessioni

Identificazione e quantificazione dei parametri delle interconnessioni

Introduzione dei modelli circuitali delle piste di interconnessione

Modelli SPICE dettagliati delle piste

Conseguenze dell'evoluzione tecnologica sulle interconnessioni

4.1 Introduzione
4.2 Un primo sguardo
4.3 Parametri delle interconnessioni: capacità, resistenza e induttanza
  4.3.1 Capacità
  4.3.2 Resistenza
  4.3.3 Induttanza
4.4 Modelli elettrici delle connessioni
  4.4.1 La connessione ideale
  4.4.2 Il modello a parametri concentrati
  4.4.3 Il modello RC a parametri concentrati
  4.4.4 Il modello rc a parametri distribuiti
  4.4.5 La linea di trasmissione
4.5 I modelli SPICE delle connessioni
  4.5.1 Le linee a parametri rc distribuiti in SPICE
  4.5.2 Le linee di trasmissione in SPICE
  4.5.3 Le prospettive: uno sguardo al futuro
4.6 Riepilogo
4.7 Approfondimenti

Materiale protetto da copyright

4.1 Introduzione

Nella storia dei circuiti integrati, le piste di interconnessione sul chip sono state sempre considerate come dei cittadini di seconda classe, prese in considerazione solo in casi speciali o per effettuare analisi ad altissima precisione. Con l'avvento delle tecnologie submicrometriche, questo scenario è cambiato rapidamente. Gli effetti parassiti associati alla piste di connessione sono caratterizzati da un tasso di riduzione diverso da quello dei dispositivi attivi, come i transistor, e quindi stanno assumendo sempre più importanza man mano che le dimensioni dei dispositivi si riducono e la velocità dei circuiti aumenta. Di fatto questi parametri cominciano a essere il fattore dominate in alcune figure di merito dei circuiti, quali velocità, consumo di potenza e robustezza. La situazione è ulteriormente aggravata dal fatto che l'evoluzione tecnologica rende fattibili chip di dimensioni sempre crescenti, con conseguente aumento della lunghezza media delle interconnessioni e quindi dei parametri parassiti a esse associate. Un'analisi attenta e approfondita dell'effetto e del comportamento delle piste di interconnessione nelle tecnologie a semiconduttore è quindi non solo desiderabile, ma essenziale.

4.2 Un primo sguardo

Il progettista di un circuito elettronico ha a disposizione diverse possibilità per realizzare le interconnessioni tra i vari dispositivi necessari a costruire il circuito. I processi allo stato dell'arte offrono diversi livelli di alluminio o rame e almeno un livello di polisilicio. Perfino le diffusioni fortemente drogate di tipo n^+ o p^+, tipicamente utilizzate per le regioni di source e drain, possono essere utilizzate per le connessioni. Queste connessioni appaiono negli schemi dei circuiti elettronici come semplici fili, senza alcun effetto apparente sulle prestazioni del circuito stesso. Dalla nostra presentazione sul processo di fabbricazione dei circuiti integrati, dovrebbe però essere evidente che questa visione è troppo semplicistica e che le connessioni dei circuiti integrati odierni formano una struttura geometrica complessa che introduce capacità, resistenze e induttanze parassite. Tutte e tre hanno diversi effetti sul comportamento di un circuito:

1. Tutte causano un aumento dei tempi di propagazione e cioè una perdita in termini di prestazioni.
2. Tutte hanno effetti in termini di dissipazione di energia e distribuzione della potenza.
3. Tutte introducono sorgenti di rumore addizionali, che degradano la robustezza del circuito.

Un progettista potrebbe decidere di agire in sicurezza, includendo tutti questi parassiti nell'analisi e nella procedura di ottimizzazione. Questo approccio conservativo non è però molto costruttivo, invero spesso non è nemmeno fattibile. In primo luogo, un modello "completo" sarebbe estremamente complesso ed è quindi applicabile solo a topologie molto ridotte ed è quindi totalmente inutile per i circuiti integrati odierni, con i loro milioni di nodi. Avrebbe inoltre lo svantaggio di rendere il sistema, per così dire, fumoso. Il comportamento

Materiale protetto da copyright

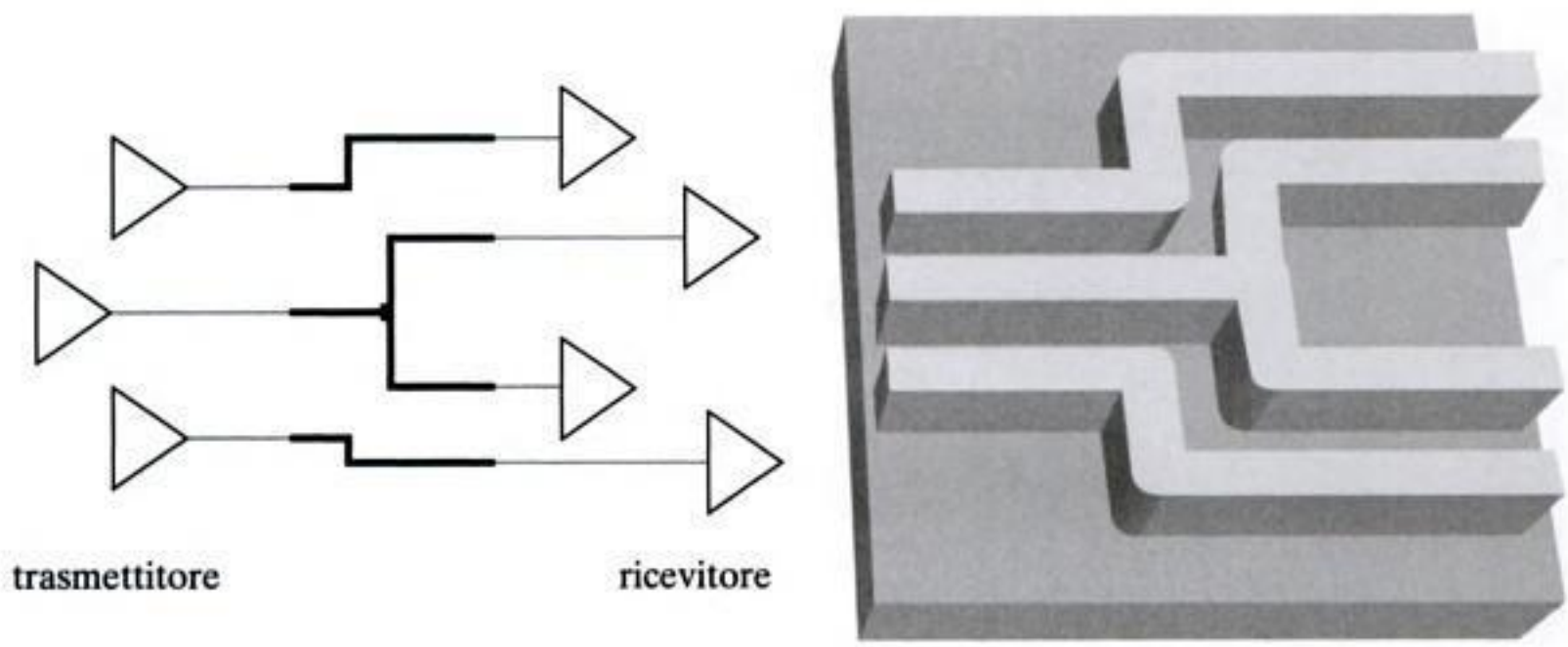


Figura 4-1 Viste schematica e fisica di una rete di connessioni. La seconda mostra solo un'area ridotta (quella in grassetto nello schema circuitale).

del circuito a uno specifico nodo è influenzato da pochi parametri; includere tutti i possibili effetti può rendere meno chiara la situazione e trasformare il processo di ottimizzazione e progettazione in una serie di tentativi, piuttosto che una ricerca illuminata e focalizzata.

Per il progettista è importante avere una chiara comprensione degli effetti parassiti delle connessioni, della loro importanza e dei relativi modelli. Ciò è ben dimostrato da un semplice esempio, come illustrato dalla Figura 4-1. Ciascuna pista in una rete di collegamenti connette un trasmettitore a un insieme di ricevitori ed è realizzata come una catena di vari segmenti di pista con diverse lunghezze e geometrie. Ipotizziamo ora che ciascun segmento sia implementato con un singolo livello di interconnessione e sia isolato dal substrato di silicio e dagli altri livelli da un strato di materiale dielettrico (occorre tener presente che nella realtà il sistema può essere ben più complicato). Un modello circuitale completo che include gli effetti delle capacità, resistenze e induttanze parassite delle connessioni è mostrato nella Figura 4-2a. Si osservi che questi elementi circuitali aggiuntivi non sono localizzati in un unico punto fisico, ma sono invece distribuiti su tutta la lunghezza della pista. Ciò è necessario quando la lunghezza della pista è significativamente maggiore della sua larghezza. Inoltre sono presenti elementi parassiti tra le varie piste, che introducono effetti di accoppiamento, non inclusi nello schema circuitale originale, tra i diversi segnali sui bus.

Analizzare il comportamento di questo schema circuitale, che rappresenta soltanto una piccola parte del circuito, è già un'attività lenta e problematica. Fortunatamente spesso è possibile apportare alcune semplificazioni sostanziali:

- Gli effetti induttivi possono essere ignorati se la resistenza della pista è sufficientemente elevata. Questo accade, per esempio, nel caso di lunghe piste di alluminio di sezione ridotta oppure se i tempi di salita e discesa dei segnali applicati sono lenti.
- Quando le piste sono corte, la sezione è elevata o il materiale utilizzato per le connessioni ha una resistività ridotta, si può utilizzare un modello esclusivamente capacitivo (vedi Figura 4-2b).

Materiale protetto da copyright

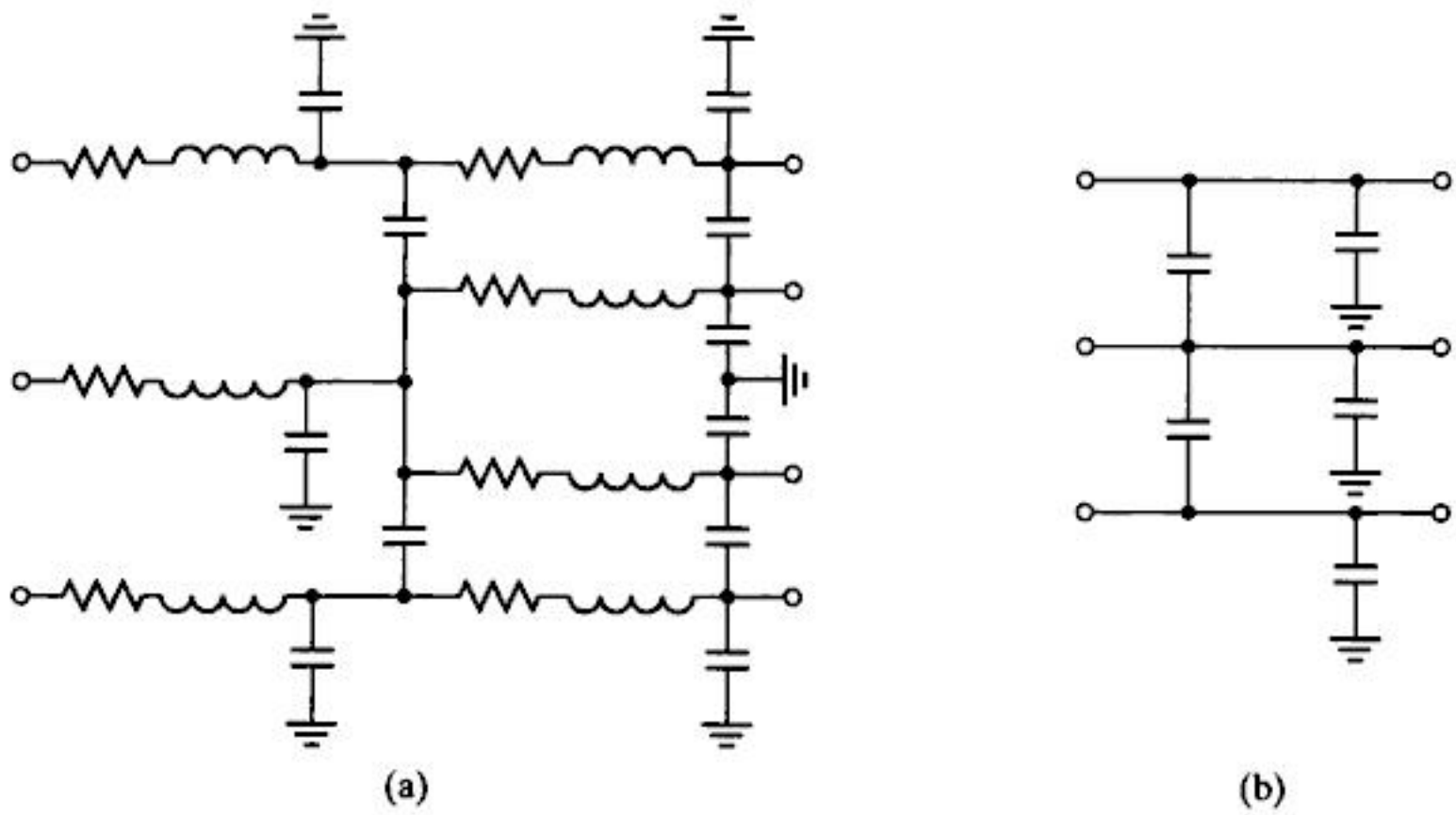


Figura 4-2 Modello delle piste per il circuito di Figura 4-1.Il modello (a) comprende la maggior parte dei parametri parassiti (con l'eccezione delle resistenze e induttanze mutue), mentre il modello (b) prende in considerazione solamente le capacità.

- Infine, quando la separazione tra piste parallele è elevata o quando le piste corrono vicine solo per una lunghezza limitata, la capacità mutue possono essere trascurate e tutte le capacità parassite possono essere rappresentate come capacità verso massa.

Ovviamente l'ultimo caso è il più semplice da modellare, analizzare e ottimizzare. Il progettista esperto è in grado di distinguere tra effetti dominanti e secondari. Questo capitolo ha lo scopo di presentare le tecniche fondamentali per stimare i valori dei parametri delle connessioni, dei semplici modelli per valutarne le conseguenze e un insieme di regole empiriche per decidere quando e dove un particolare modello o effetto debba essere preso in considerazione.

4.3 Parametri delle interconnessioni: capacità, resistenza e induttanza

4.3.1 Capacità

Una modellazione accurata della capacità delle piste di connessione nei circuiti integrati allo stato dell'arte è un compito tutt'altro che semplice ed è tuttora oggetto di ricerche avanzate. La situazione è ulteriormente complicata dal fatto che i circuiti integrati contemporanei sono una struttura tridimensionale, come mostra chiaramente la sezione trasversale illustrata dalla Figura 2-8. La capacità di una pista è funzione della sua forma, degli altri elementi circuitali vicini, della sua distanza dal substrato e della distanza dalle piste vicine. Piuttosto che perdersi in equazioni e modelli complicati, un progettista tipicamente utilizza uno strumento di estrazione avanzato per ottenere un valore preciso della capacità di interconnessione in un layout completo. Molti costruttori di semiconduttori forniscono anche dati sperimentali per i vari contributi capacitivi, in base alle misure effettuate su un certo numero di chip. Di fatto,

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Tabella 4-1 Permettività relativa di alcuni tipici materiali dielettrici.

Materiale	ε_r
Vuoto	1
Aerogel	~1.5
Polimeri (organici)	3-4
Diossido di silicio	3.9
Vetro-epossidico (schede)	5
Nitruro di silicio (Si_3N_4)	7.5
Alluminio (package)	9.5
Silicio	11.7

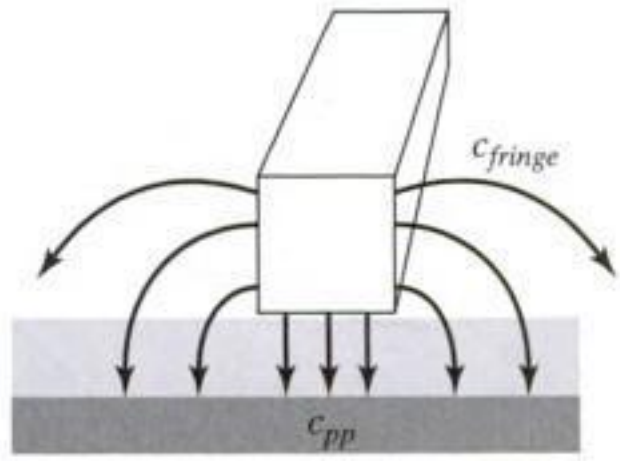
anni, abbiamo assistito a una riduzione costante del rapporto *W/H*, che nel caso dei processi più avanzati può addirittura essere minore di uno. Questa situazione è chiaramente osservabile nella sezione trasversale della Figura 2-8. In queste condizioni, il modello a piatti piani paralleli diviene inaccurato. La capacità tra i bordi della pista e il substrato, chiamata *capacità di bordo*, non può essere ignorata dato che contribuisce alla capacità totale; questo effetto è mostrato nella Figura 4-4a. Ottenere un modello accurato per tale geometria è difficile, quindi, come raccomanda l'esperienza ingegneristica, utilizziamo un modello semplificato che approssima la capacità totale come somma di due contributi (Figura 4-4b): un contributo di area, determinato dal campo ortogonale tra una pista di larghezza *w* e il piano di massa, in parallelo con la capacità di bordo, rappresentata da una pista cilindrica di dimensione pari allo spessore dell'interconnessione *H*. Il modello approssimato che ne risulta è semplice ed efficace:

$$c_{pista} = c_{pp} + c_{bordo} = \frac{w\varepsilon_{di}}{t_{di}} + \frac{2\pi\varepsilon_{di}}{\log(2t_{di}/H+1)} \tag{4.2}$$

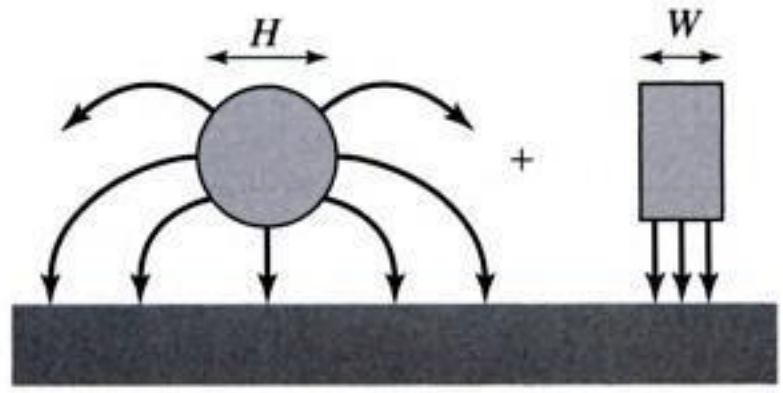
dove $w = W - H/2$ è una buona approssimazione per la larghezza del condensatore a piatti paralleli. Nel corso del tempo sono stati proposti molti modelli accurati (per esempio, [Vdmeijs84]), che però tendono a essere decisamente più complessi e vanno oltre gli scopi di questa discussione.

Al fine di illustrare l'importanza della componente di bordo, la Figura 4-5 riporta la capacità di bordo in funzione di (W/t_{di}) o, indirettamente, di (W/H). Per valori elevati di (W/H), la capacità totale è approssimabile dal solo contributo di area. Per (W/H) minore di 1.5, la componente di bordo è quella dominante. La capacità di bordo può far aumentare la capacità totale di un fattore maggiore di 10 nel caso di piste di larghezza ridotta. Risulta interessante osservare che il valore totale di capacità si appiattisce al valore costante di

Materiale protetto da copyright



(a) Campo di bordo



(b) Modello della capacità di bordo

Figura 4-4 La capacità di bordo. Il modello scompone la capacità in due contributi: un condesatore a facce piane parallele e una capacità di bordo, modellata da una pista a sezione cilindrica con un diametro pari allo spessore della pista.

circa 1 pF/cm per larghezze della pista minori dello spessore dell'isolante; in altre parole, la capacità non è più funzione della larghezza.

Fin qui abbiamo limitato la nostra analisi al caso di un singolo conduttore rettangolare posto al di sopra di un piano di massa. Tale struttura (detta *microstrip*) era un ottimo modello per le interconnessioni nei circuiti a semiconduttore, finché i livelli di interconnessione erano limitati a 1 o 2. I processi contemporanei, però, mettono a disposizione molti più livelli, per di più integrati con una maggiore densità. In tale contesto, l'ipotesi che una pista sia completamente isolata da tutto ciò che la circonda, a meno di un accoppiamento capacitivo con la massa, diviene irrealistica. Questa è la situazione mostrata nella Figura 4-6, dove sono identificate le componenti capacitive di una pista inclusa in una gerarchia di interconnessione. Ciascuna pista non è accoppiata capacitivamente al solo substrato a massa, ma anche alle altre piste vicine, sullo stesso livello o su quelli adiacenti. In un'analisi al primo ordine, questa situazione non altera effettivamente il valore della capacità totale connessa alla pista presa in considerazione. La differenza fondamentale sta però nel fatto che tale capacità non è più per intero riferita a massa- La maggior parte di tali contributi capacitivi sono infatti connessi ad altre piste, le quali hanno livelli di tensione che variano dinamicamente. Più avanti vedremo come tali contributi *capacitivi flottanti*, non solo sono una fonte di rumore (accoppiamenti mutui, *cross talk*), ma possono addirittura degradare le prestazioni dinamiche del circuito.

In definitiva si può affermare che le capacità mutue tra le varie piste stanno divenendo il fattore parassita dominante nelle strutture con più livelli di interconnessione. Tale effetto

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Capacità totale: 11 pF

Notiamo il fattore 2 nel calcolo della capacità di bordo, dovuto al fatto che si prendono in considerazione entrambi i lati della pista.

Supponiamo ora che venga posta una seconda pista di fianco alla prima, separata dalla sola distanza minima. Dalla Tabella 4-3, possiamo ricavare che questa pista si accoppierà alla prima per mezzo di una capacità pari a

$$C_{mutua} = (0.1 \times 10^6\ \mu\text{m}) \times 95\ \text{aF}/\mu\text{m} = 9.5\ \text{pF}$$

che è quasi pari alla capacità totale verso massa.

Un esercizio simile potrebbe dimostrare che spostare le due piste al livello Al-4, ridurrebbe la capacità verso massa a 3.45 pF (0.65 pF di area e 2.8 pF di bordo), mentre la capacità mutua rimarrebbe circa la stessa, cioè 8.5 pF.

4.3.2 Resistenza

La resistenza di una pista è proporzionale alla sua lunghezza L, e inversamente proporzionale all'area della sua sezione A. La resistenza di un conduttore di forma rettangolare, come quello mostrato in Figura 4-3, può essere espressa come

$$R = \frac{\rho L}{A} = \frac{\rho L}{HW} \tag{4.3}$$

dove la costante ρ è la resistività del materiale (in Ω m). La resistività di alcuni materiali comunemente utilizzati come conduttori è riportata nella Tabella 4-4. L' alluminio è il materiale di interconnessione utilizzato più spesso nei circuiti integrati, grazie al suo costo ridotto e alla compatibilità con i normali processi di fabbricazione di circuiti integrati. Sfortunatamente però ha una resistività piuttosto elevata, rispetto a quella di altri materiali, come per esempio il rame. Data la richiesta di prestazioni sempre più spinte, il rame sta divenendo il materiale prescelto come conduttore nei processi più avanzati.

Poiché H per una data tecnologia è una costante, l'Eq. (4.3) può essere riscritta come

$$R = R_{\square} \frac{L}{W} \tag{4.4}$$

Tabella 4-4 Resistività di conduttori utilizzati abitualmente (a 20°C).

Materiale	ρ (Ω-m)
Argento (Ag)	1.6×10^{-8}
Rame (Cu)	1.7×10^{-8}
Oro (Au)	2.2×10^{-8}
Alluminio (Al)	2.7×10^{-8}
Tungsteno (W)	5.5×10^{-8}

Materiale protetto da copyright

dove

$$R_q = \frac{\rho}{H} \tag{4.5}$$

è la *resistenza quadro* o *resistenza superficiale (sheet resistance)* del materiale, espressa in Ω/quadro. Tale parametro mette in evidenza che la resistenza di una pista di forma quadrata di un dato materiale è indipendente dal valore del suo lato, com'è evidente dall'Eq. (4.4). Per ottenre la resistenza di un pista è quindi sufficiente moltiplicare la resistenza quadro per il rapporto *L/W*.

Dati caratteristici della resistenza delle interconnessioni

I valori tipici per le resistenze quadro dei vari materiali di interconnessione sono riportati nella Tabella 4-5.

Da questa tabella possiamo facilmente dedurre che l'alluminio è il materiale da preferire per realizzare lunghe interconnessioni, mentre il polisilicio andrebbe utilizzato esclusivamente per connessioni locali. Sebbene la resistenza quadro delle zone diffuse (n^+, p^+) sia comparabile con quella del polisilicio, è sconsigliabile realizzare interconnessioni mediante tali diffusioni a causa dell'elevata capacità parassita.

Tabella 4-5 Valori della resistenza quadro per un tipico processo CMOS da 0.25 μm.

Materiale	resistenza quadro (Ω/q)
Diffusioni dell well n o p	1000–1500
Diffusioni n^+, p^+	50–150
Diffusioni ricoperte con siliciuro n^+, p^+	3–5
Polisilicio n^+, p^+	150–200
Polisilicio ricoperto con siliciuro n^+, p^+	4–5
Alluminio	0.05–0.1

■

I processi più avanzati mettono anche a disposizione livelli di polisilicio e di diffusione trattati con siliciuri, che sono composti da silicio e altri materiali refrattari. Questo permette di ottenere materiali ad alta conducibilità, che possono sopportare le temperature elevate di alcuni passi del processo, senza fondere. Alcuni esempi di siliciuri sono WSi_2, $TiSi_2$, $PtSi_2$, e TaSi; per esempio il WSi_2 ha una resistività ρ pari a 130 μΩ-cm, che è circa otto volte minore di quella del polisilicio. I siliciuri sono utilizzati per la maggior parte dei casi in configurazioni dette *polycide*, che sono semplicemente la combinazione di un livello di polisilicio (*polisilicio*) e di uno di siliciuro (*silicide*). Un esempio tipico è uno strato inferiore di polisilicio con un rivestimento superiore di siliciuro; ciò combina le proprietà migliori dei due materiali: buona aderenza e copertura (dal polisilicio) e alta conducibilità (dal siliciuro). La Figura 4-8 mostra un transistor MOSFET fabbricato utilizzando un gate in polycide: il van-

Materiale protetto da copyright

taggio di avere un gate trattato con siliciuro sta nella ridotta resistenza di gate; similmente regioni di source e drain trattate con siliciuro implicano la riduzione delle resistenze al source e al drain del dispositivo.

Il passaggio da un livello di interconnessione a un altro introduce ulteriore resistenza, chiamata *resistenza di contatto*. La migliore strategia nel progettare le interconnessioni consiste quindi nel mantenere le piste che portano i segnali il più possibile su un unico livello, evitando contatti o *via* non indispensabili. Un'altra possibilità per ridurre la resistenza di contatto è quella di aumentare l'area dei fori di contatto, ma sfortunatamente in tal caso la corrente tende a concentrarsi sui bordi del contatto stesso. Questo fenomeno, chiamato *current crowding*, pone in pratica un limite superiore alla dimensione del contatto. Valori tipici della resistenza di contatto (nel caso di contatti a dimensione minima) per un processo da 0.25 μm sono: 5–20 Ω da metallo o polisilicio a n^+, p^+, e da metallo a polisilicio; 1–5 Ω per via (contatti metallo-metallo).

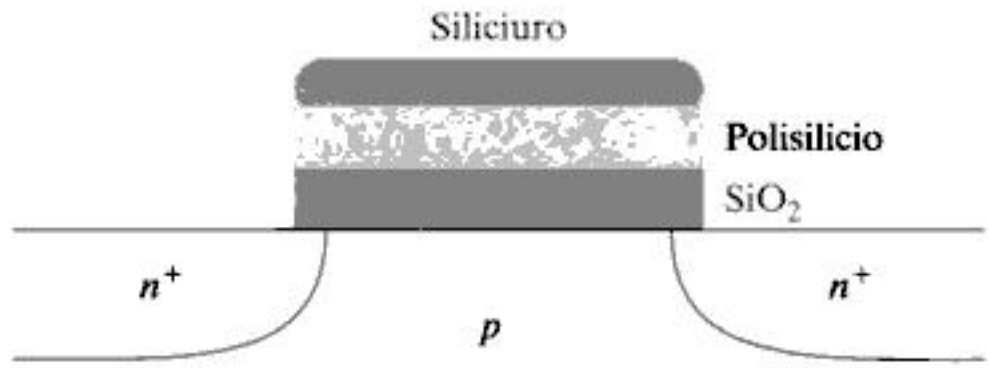


Figura 4-8 Un MOSFET con gate trattato al siliciuro.

Esempio 4.2 Resistenza di una pista metallica

Consideriamo nuovamente la pista in alluminio dell'Esempio 4.1, che è lunga 10 cm, larga 1 μm e realizzata con il primo livello di alluminio. Ipotizzando una resistenza quadro per l'alluminio pari a 0.075 Ω/q, possiamo calcolare la resistenza totale come segue:

$$R_{pista} = 0.075\ \Omega/\text{q} \times (0.1 \times 10^6\ \mu\text{m}) / (1\ \mu\text{m}) = 7.5\ \text{k}\Omega$$

Realizzare la pista in polisilicio, con una resistenza quadro pari a 175 Ω/quadro comporta un aumento della resistenza totale fino a 17.5 MΩ, valore ovviamente inaccettabile. Il polisilicio trattato con siliciuro, con una resistenza quadro da 4 Ω/quadro rappresenta un'alternativa migliore, ma comunque si traduce in una pista con resistenza parassita pari a 400-kΩ.

Finora abbiamo supposto che la resistenza di una pista sia lineare e costante, fatto confermato nella maggior parte dei circuiti a semiconduttore. A frequenze molto elevate, però, entra in gioco un altro fenomeno, noto come *effetto pelle*, che rende la resistenza dipendente dalla frequenza. Le correnti ad alta frequenza tendono a fluire maggiormente lungo la superficie del conduttore, con una densità di corrente che decade esponenzialmente in funzione della profondità nel conduttore. La *profondità caratteristica* (*skin depth*) δ è definita come

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

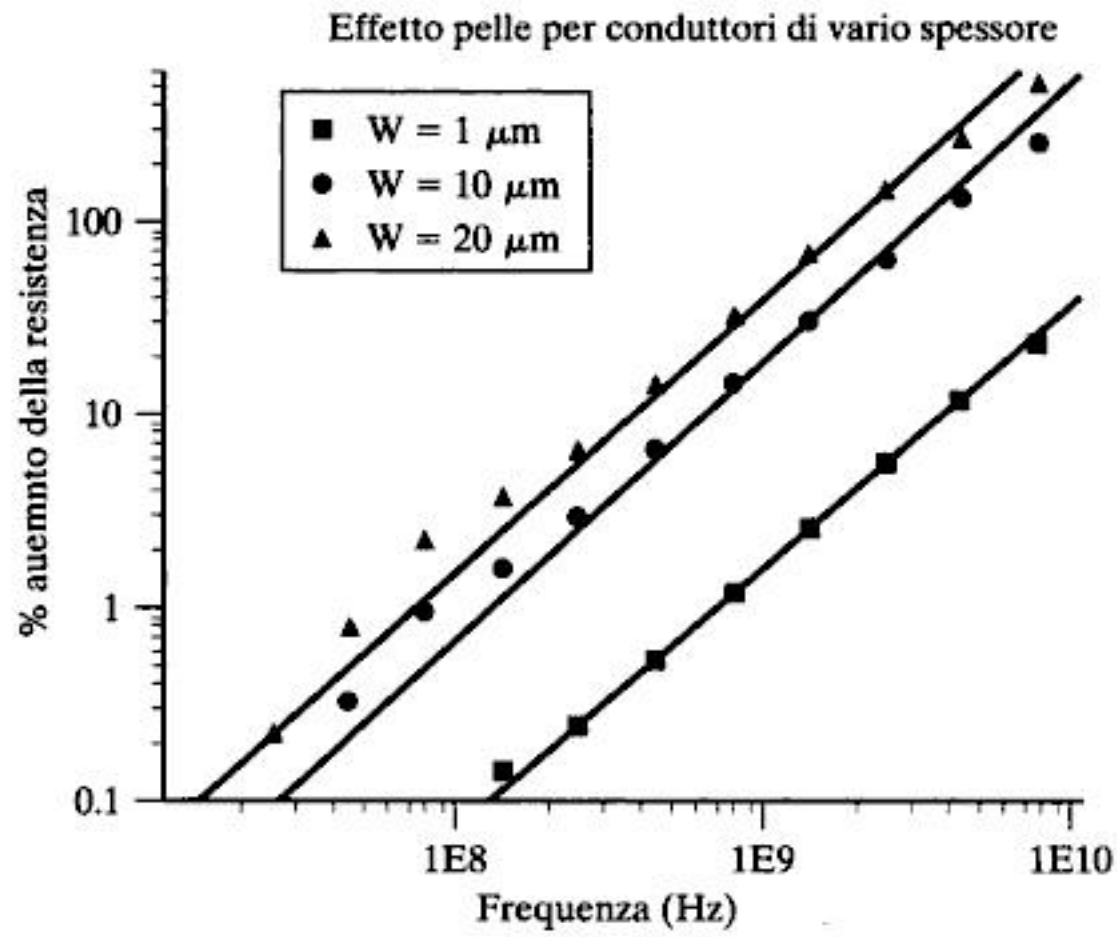


Figura 4-10 Aumento della resistenza in seguito all'effetto pelle in funzione della frequenza e dello spessore della pista. Tutte le simulazioni sono state effettuate per una pista di spessore 0.7 μm [Sylvester97].

dimensione maggiore della pista deve essere almeno di 5.2 μm affinché l'effetto sia significativo a 1 GHz. Questo è confermato dai risultati più accurati della simulazione di Figura 4-10, che riporta l'aumento della resistenza per effetto pelle per conduttori di alluminio di larghezze diverse. Si può osservare un aumento del 30% della resistenza a 1 GHz per una pista da 20 μm, mentre l'aumento nel caso di un pista da 1 μm è meno del 2%.

Per riassumere, l'effetto pelle è effettivamente un problema solo per le piste più larghe. Dato che i clock tendono a essere i segnali a maggiore frequenza in un chip e sono anche distribuiti con linee piuttosto spesse, per limitarne la resistenza, probabilmente l'effetto pelle sarà significativo per tali connessioni. Questo può effettivamente diventare un problema serio per sistemi che operano a qualche GHz, per i quali i clock definiscono le prestazioni globali del chip (periodo di ogni ciclo, istruzioni per secondo, ecc.). Un altro aspetto importante è che l'utilizzo di conduttori migliori, quali il rame, può rendere significativo l'effetto pelle a frequenza minori.

4.3.3 Induttanza

I progettisti di circuiti integrati tendono a liquidare l'induttanza come un concetto sentito nei corsi di fisica, ma che non ha nessun impatto nel proprio campo. Anche se ciò era sicuramente vero nei primi decenni della progettazione di circuiti integrati digitali, anche con l'utilizzo di materiali di interconnessione a bassa resistenza e con l'aumento della frequenza di commutazione al di sopra del GHz, gli effetti induttivi cominciano a divenire significativi

Materiale protetto da copyright

perfino all'interno dei chip. Le conseguenze di induttanze sul chip comprendono effetti di oscillazione e sovratensioni, riflessione di segnali a causa di impedenze non adattate, accoppiamenti induttivi tra linee e rumore di commutazione dovuto a cadute di tensione ($L \cdot di/dt$).

L'induttanza di un elemento circuitale può sempre essere stimata sfruttando la sua definizione, per cui una variazione nella corrente che fluisce in tale elemento causa una caduta di tensione pari a

$$\Delta V = L \frac{di}{dt} \tag{4.9}$$

Quindi è possibile calcolare l'induttanza di una pista direttamente sulla base delle sue caratteristiche geometriche e dei materiali utilizzati. Un approccio semplificato si basa sulla proprietà per cui la capacità c e l'induttanza l (per unità di lunghezza) di una pista rispettano la relazione

$$cl = \varepsilon\mu \tag{4.10}$$

dove ε e μ sono rispettivamente la permettività e la permeabilità del dielettrico che circonda la pista. L'avvertenza è che il conduttore deve essere completamente circondato da un dielettrico omogeneo, perché tale espressione sia valida; situazione spesso non verificata. Tuttavia quando il conduttore è circondato da materiali dielettrici diversi, è sempre possibile utilizzare un valore medio per le costanti dielettriche, in modo che l'Eq. (4.10) possa comunque essere utilizzata per stimare il valore dell'induttanza.

Possiamo mettere in evidenza qualche altra relazione interessante, ricavata delle equazioni di Maxwell. Il prodotto costante di permeabilità e permettività definisce anche la velocità ν a cui un'onda elettromagnetica si propaga nel mezzo, che è data da

$$\nu = \frac{1}{\sqrt{lc}} = \frac{1}{\sqrt{\varepsilon\mu}} = \frac{c_0}{\sqrt{\varepsilon_r \mu_r}} \tag{4.11}$$

dove c_0 è uguale alla velocità della luce nel vuoto (30 cm/ns). La velocità di propagazione per un certo numero di materiali utilizzati per la fabbricazione di circuiti elettronici è riportata nella Tabella 4-6. La propagazione in SiO_2 è due volte più lenta di quella nel vuoto.

Tabella 4-6 Costanti dielettriche e velocità di propagazione d'onda per diversi ma-teriali utilizzati nella fabbricazione di circuiti elettronici. La permeabilità relativa μ_r della maggior parte dei dielettrici è circa unitaria.

Dielettrico	ε_r	Velocità di propagazione (cm/ns)
Vuoto	1	30
SiO_2	3.9	15
Scheda (vetro epossidico)	5.0	13
Alluminio (package ceramico)	9.5	10

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

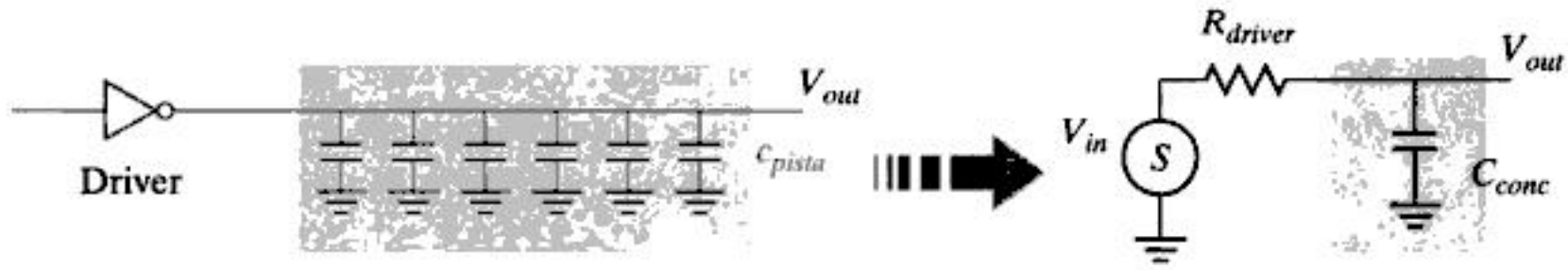


Figura 4-11 Modello capacitivo distribuito e concentrato di una pista. $C_{conc} = L \times c_{pista}$, dove L è la lunghezza della pista e c_{pista} la capacità per unità di lunghezza. Il driver è rappresentato come un generatore di tensione con una resistenza serie R_{driver}.

Nell'Esempio 4.1, abbiamo calcolato che la capacità totale concentrata per questa pista vale 11 pF.

Il funzionamento di questa semplice rete RC è descritto dalla seguente equazione differenziale (simile all'espressione ricavata nell'Esempio 1):

$$C_{conc} \frac{dV_{out}}{dt} + \frac{V_{out} - V_{in}}{R_{driver}} = 0$$

Quando si applica in ingresso un gradino (con V_{in} che commuta da 0 a V), è noto che la risposta dinamica del circuito è una funzione esponenziale, data dall'espressione

$$V_{out}(t) = (1 - e^{-t/\tau})\, V$$

dove $\tau = R_{driver} \times C_{conc}$ è la costante di tempo della rete.

Il tempo per raggiungere il 50% dell'escursione logica è $t = \ln(2)\tau = 0.69\tau$. Similmente è necessario un tempo $t = \ln(9)\tau = 2.2\tau$ per andare dal 10% al 90% dell'escursione. Inserendo i valori numerici di questo particolare esempio, calcoliamo

$$t_{50\%} = 0.69 \times 10\ \text{K}\Omega \times 11\ \text{pF} = 76\ \text{ns}$$
$$t_{90\%} = 2.2 \times 10\ \text{K}\Omega \times 11\ \text{pF} = 242\ \text{ns}$$

Questi valori non sono accettabili nemmeno per i circuiti digitali di prestazioni più basse. Nei Capitoli 5 e 9 discuteremo le tecniche per superare tale collo di bottiglia, quali la riduzione della resistenza di uscita del circuito che pilota la pista.

Sebbene il modello a capacità concentrate sia il più utilizzato, qualche volta è necessario introdurre modelli concentrati di una linea resistiva o induttiva. Questo è spesso il caso dello studio della rete di distribuzione delle alimentazioni. Sia la resistenza sia l'induttanza delle piste di alimentazione possono essere interpretate come sorgenti parassite di rumore che causano cadute e oscillazioni di tensione sulle piste di alimentazione.

4.4.3 Il modello *RC* a parametri concentrati

Le piste metalliche con lunghezze maggiori di pochi millimetri hanno una resistenza significativa, quindi l'ipotesi di equipotenzialità, presentata con il modello a capacità concentrate, non è più realistica e si rende necessario un modello resistivo-capacitivo.

Un primo approccio concentra la resistenza di ogni segmento della pista in una singola componente totale R e, similmente, combina tutti i contributi capacitivi in un condesatore

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

- Il ritardo di una linea *rc* distribuita è **la metà del ritardo** che il modello *RC* concentrato avrebbe predetto. Quest'ultimo si ottiene concentrando la resistenza e la capacità totali in elementi unici e ha una costante di tempo pari a *RC* (come anche si ottiene ponendo $N = 1$ nell'Eq. (4.15)). Questo conferma l'osservazione fatta in precedenza che il modello concentrato rappresenta una stima pessimistica del ritardo della pista resistiva.

ATTENZIONE: Bisogna tenere presente che una catena *RC* è caratterizzata da un certo numero di costanti di tempo. La formula di Elmore determina solamente il valore di quella dominante e quindi è solo un'approssimazione del primo ordine.

La formula del ritardo di Elmore si è dimostrata estremamente utile. Oltre a permettere l'analisi delle interconnessioni, la formula può essere utilizzata anche per approssimare il ritardo di propagazione in reti di transistor complesse. Nel modello a interruttore, i transistor sono sostituiti dalla loro resistenza equivalente linearizzata. La stima del tempo di propagazione si riduce quindi all'analisi della rete *RC* equivalente.

Sono anche stati definiti limiti massimi e minimi per le forme d'onda delle tensioni in una rete *RC* [Rubinstein83]. Questi vincoli costituiscono la base della maggior parte degli strumenti di analisi al calcolatore a livello di interruttore o funzionale [Horowitz83]. Un risultato interessante [Lin84] è che la forma d'onda esponenziale con la costante di tempo pari al ritardo di Elmore è sempre compresa tra questi limiti massimo e minimo, il che dimostra la validità dell'approssimazione di Elmore.

4.4.4 Il modello *rc* a parametri distribuiti

Nel paragrafo precedente, abbiamo visto come il modello *RC* distribuito sia un'approssimazione pessimistica di una linea resistiva–capacitiva e come un modello *rc* distribuito (Figura 4-14a) sia più appropriato. Come in precedenza, L rappresenta la lunghezza totale della pista, mentre r e c rappresentano la resistenza e la capacità per unità di lunghezza. Una rappresentazione schematica della linea *rc* distribuita è riportata in Figura 4-14b.

La tensione al nodo i di questa rete può essere calcolata risolvendo il seguente sistema di equazioni alle derivate parziali:

$$c\Delta L\frac{\partial V_i}{\partial t} = \frac{(V_{i+1} - V_i) + (V_{i-1} - V_i)}{r\Delta L} \tag{4.17}$$

Il comportamento corretto della linea *rc* distribuita si ottiene quindi riducendo ΔL asintoticamente a 0. Per $\Delta L \to 0$, l'Eq. (4.17) diviene la ben nota *equazione di diffusione*:

$$rc\frac{\partial V}{\partial t} = \frac{\partial^2 V}{\partial x^2} \tag{4.18}$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

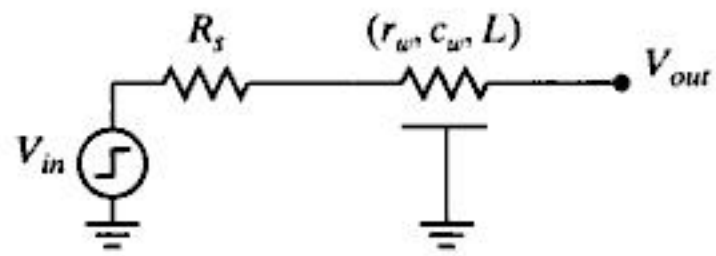


Figura 4-16 Linea *rc* di lunghezza *L* pilotata da una sorgente con resistenza serie R_s.

sorgente con una resistenza serie pari a 1 kΩ e una pista in Al-1 larga 1 μm (r = 0.075 Ω/μm). Si ottiene così una lunghezza critica di 2.67 cm.

4.4.5 La linea di trasmissione

Quando la velocità di commutazione dei circuiti è sufficientemente elevata e il materiale di interconnessione ha una resistenza abbastanza bassa, l'induttanza della pista diventa il parametro dominante sul ritardo ed è quindi necessario prendere in considerazione effetti di linea di trasmissione. Più precisamente, questo succede quando i tempi di salita e discesa del segnale sono comparabili al tempo di propagazione della forma d'onda lungo la linea. Con l'avvento delle connessioni in rame e l'elevata velocità di commutazione derivante dalle tecnologie submicrometriche, nei sistemi CMOS più veloci le piste devono essere trattate come linee di trasmissione.

In questo paragrafo analizziamo dapprima un modello per la linea di trasmissione; di seguito applichiamo tale modello a una tecnologia a semiconduttore attuale e identifichiamo quali effetti devono essere considerati nel flusso di progetto.

Modello della linea di trasmissione

Similmente alla resistenza e alla capacità di una linea di interconnessione, anche l'induttanza è distribuita lungo la pista. Un modello *rlc* distribuito di una pista, conosciuto come *modello a linea di trasmissione*, è l'approssimazione più accurata del comportamento effettivo. La linea di trasmissione ha la proprietà fondamentale di propagare un segnale lungo il mezzo come un'*onda*, al contrario di quanto accade del modello *rc* distribuito, dove il segnale *diffonde* dalla sorgente alla destinazione, secondo l'Eq. (4.18), l'equazione di diffusione. Come un'onda si propaga in un mezzo trasferendo alternativamente energia dal campo elettrico a quello magnetico, il segnale si propaga in una linea trasferendo energia dalla componente capacitiva a quella induttiva.

Consideriamo il punto x lungo la linea di trasmissione di Figura 4-17 all'istante t; vale il seguente sistema di equazioni:

$$\begin{aligned} \frac{\partial v}{\partial x} &= -ri - l\frac{\partial i}{\partial t} \\ \frac{\partial i}{\partial x} &= -gv - c\frac{\partial v}{\partial t} \end{aligned} \tag{4.22}$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

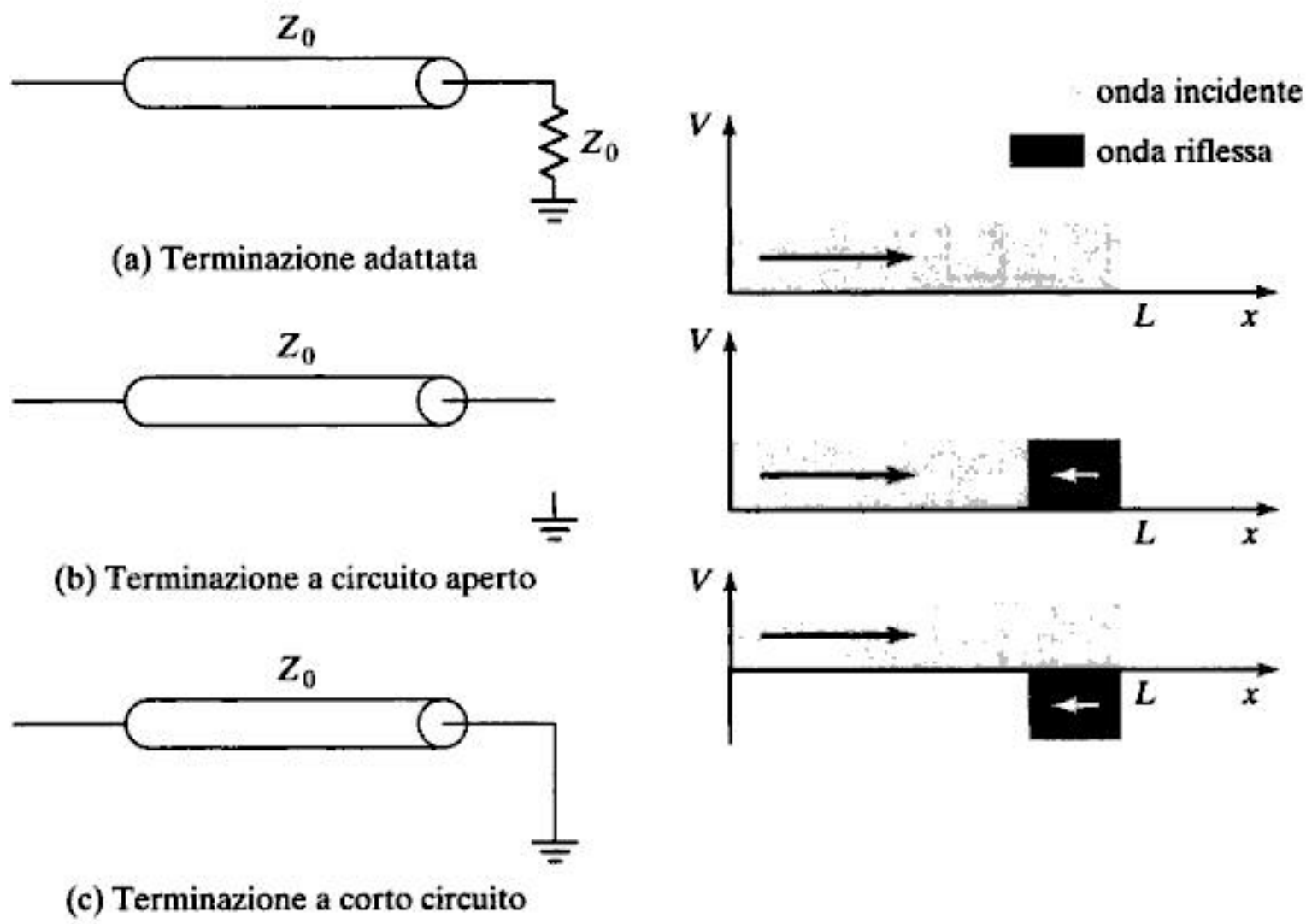


Figura 4-19 Comportamento di differenti terminazioni per linee d'onda.

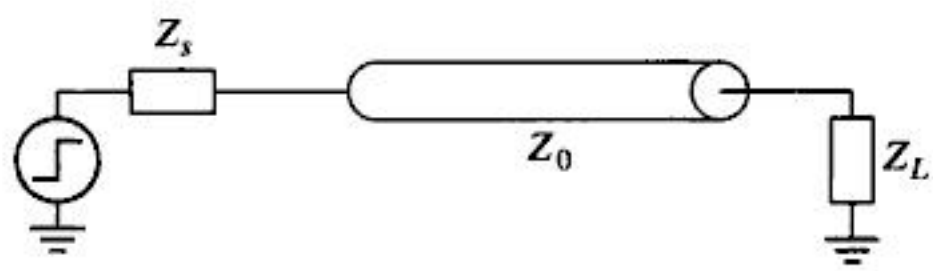


Figura 4-20 Linea di trasmissione con impedenza di terminazione.

dall'impedenza serie della sorgente Z_S e dall'impedenza di carico Z_L alla fine della linea, come mostrato nella Figura 4-20.

Consideriamo dapprima il caso in cui l'estremo della linea sia un circuito aperto, cioè $Z_L = \infty$, e $\rho_L = 1$. Un'onda entrante viene riflessa completamente, senza inversione della fase. Nell'ipotesi che l'impedenza di sorgente sia resistiva, tre possibili situazioni sono rappresentate nella Figura 4-21: $R_S = 5Z_0$, $R_S = Z_0$, e $R_S = 1/5Z_0$.

1. **Resistenza di sorgente elevata**—$R_S = 5Z_0$ (Figura 4-21a)

Solamente una piccola frazione del segnale di ingresso V_{in} da 5 V viene iniettata nella linea di trasmissione. La quantità iniettata è determinata dal partitore resistivo formato dalla resistenza di sorgente e dall'impedenza caratteristica Z_0:

$$V_{sorgente} = (Z_0 / (Z_0 + R_S))\, V_{in} = 1/6 \times 5 \text{ V} = 0.83 \text{ V} \tag{4.33}$$

Questo segnale raggiunge l'estremità della linea dopo L/υ secondi, dove L indica la lunghezza della pista, e viene quindi completamente riflesso, di fatto raddoppiando l'ampiezza dell'onda ($V_{dest} = 1.67$ V). Il tempo necessario all'onda per propagarsi da un'estremità

Materiale protetto da copyright

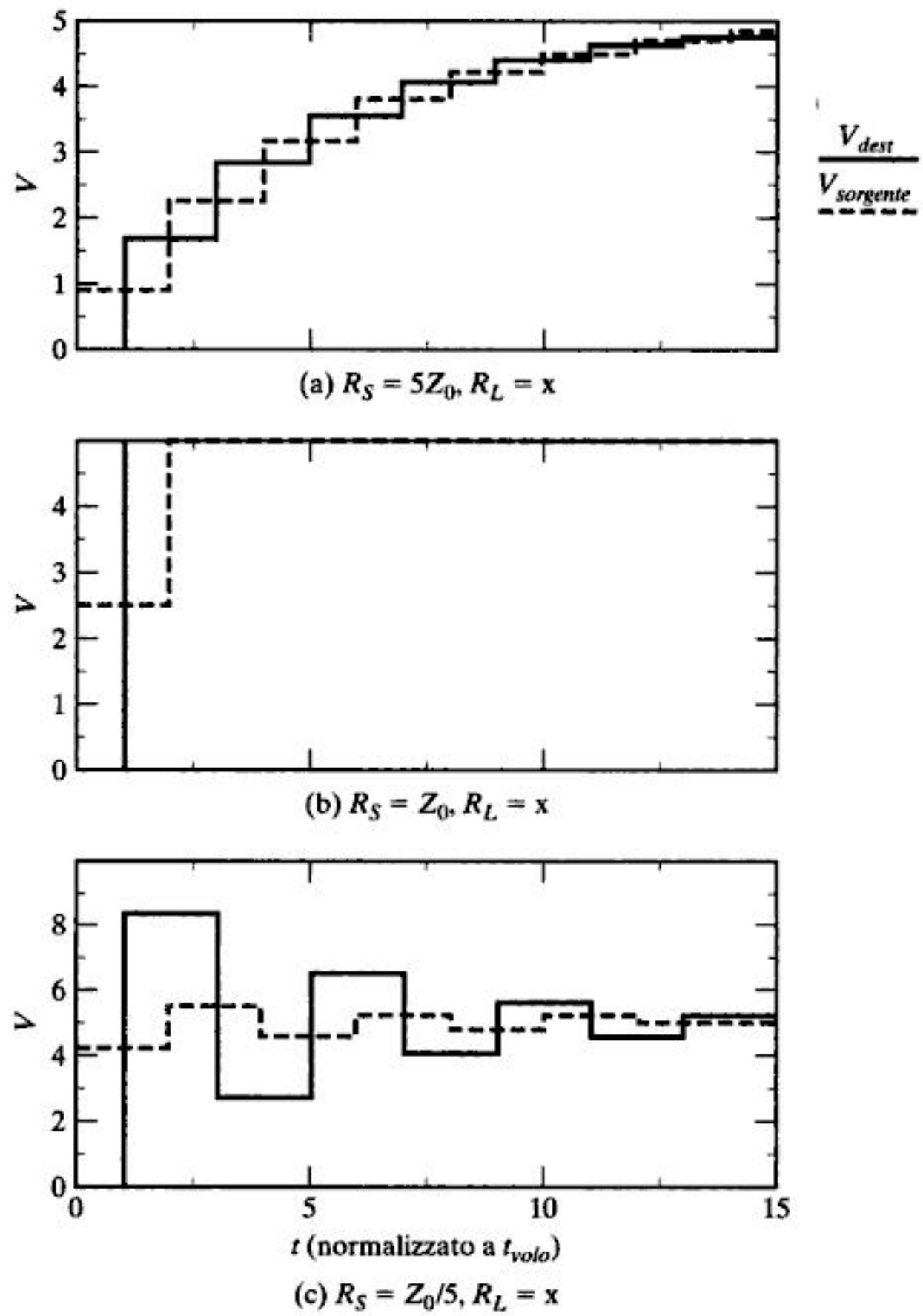


Figura 4-21 Risposta al transitorio della linea di trasmissione.

all'altra della linea è chiamato *tempo di volo*, $t_{volo} = L/\upsilon$. Un fenomeno simile si manifesta quando l'onda raggiunge nuovamente la sorgente: l'onda incidente viene riflessa con un'ampiezza determinata dal coefficiente di riflessione, che in questo particolare caso vale due terzi:

$$\rho_S = \frac{5Z_0 - Z_0}{5Z_0 + Z_0} = \frac{2}{3} \tag{4.34}$$

Materiale protetto da copyright

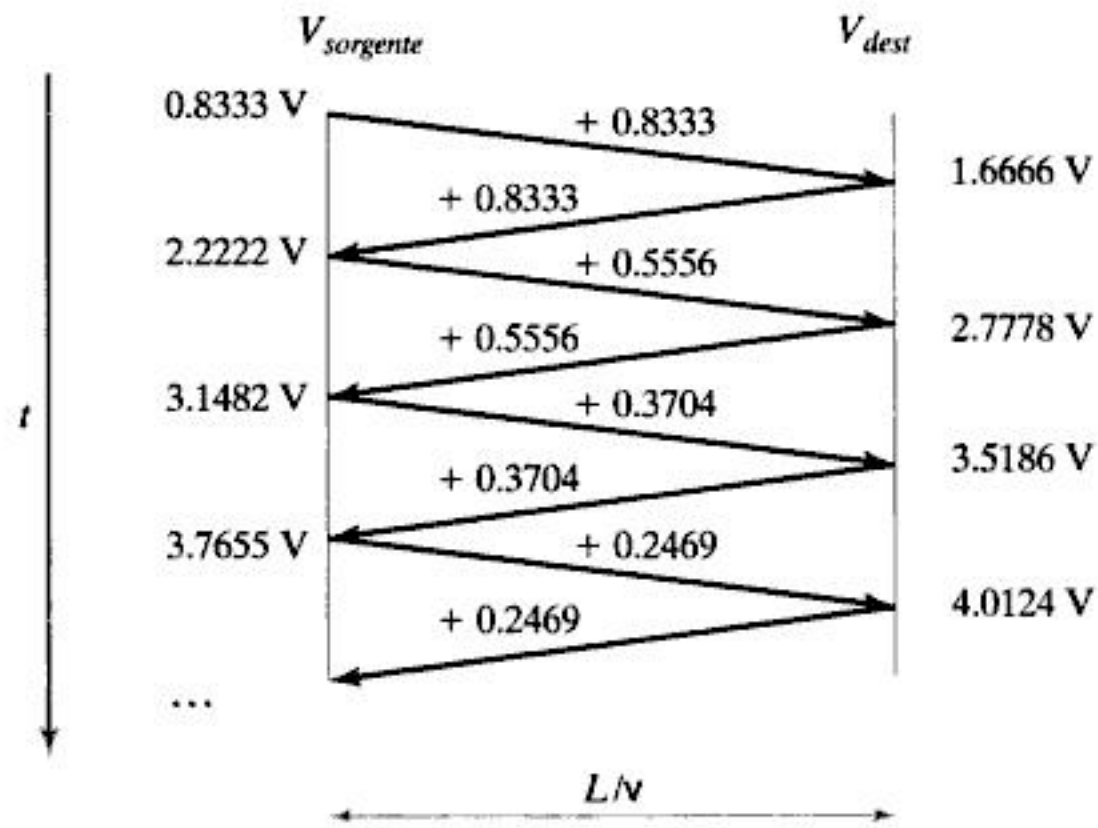


Figura 4-22 Diagramma a traliccio per $R_S = 5Z_0$ e $RL = \infty$. $V_{step} = 5$ V (come nella Figura 4-21a).

L'ampiezza della tensione al nodo sorgente e di destinazione raggiunge gradualmente il valore finale di V_{in}, ma il tempo di salita totale è di molte volte superiore a L/υ.

Quando, come in questo caso, sono presenti riflessioni multiple è difficile tenere traccia delle varie onde lungo la linea e dei livelli di tensione totali. Spesso si utilizza una costruzione grafica, chiamata *diagramma a traliccio*, per tenere traccia dell'evoluzione di tali dati (vedi Figura 4-22). Il diagramma contiene i valori della tensione al nodo sorgente e a quello di destinazione, così come i valori delle forme d'onda incidenti e riflesse. La tensione della linea al punto di terminazione è uguale alla somma della tensione precedente, dell'onda incidente e di quella riflessa.

2. **Resistenza di sorgente piccola**—$R_S = Z_0/5$ (Figura 4-21c)

Una porzione cospicua dell'ingresso è iniettata nella linea e l'ampiezza del segnale raddoppia all'estremità di destinazione, causando una notevole sovratensione, mentre dal lato della sorgente, la fase del segnale è invertita ($\rho_S = -2/3$). Il segnale quindi rimbalza avanti e indietro nella linea, inducendo un'oscillazione evidente; sono necessari diversi multipli di L/υ perché si stabilizzi.

3. **Resistenza di sorgente adattata**—$R_S = Z_0$ (Figura 4-21b)

Metà del segnale di ingresso viene iniettato alla sorgente, mentre all'estremità di destinazione il segnale viene raddoppiato dalla riflessione e il valore finale viene raggiunto immediatamente. Appare evidente che questa è la configurazione più efficace.

Si noti che l'analisi fin qui svolta è ideale poiché assume che il segnale di ingresso abbia tempo di salita nullo. In condizioni reali, i segnali sono sostanzialmente più graduali, come mostrato nella risposta simulata della Figura 4-23 (per $R_S = Z_0/5$ e $t_r = t_{volo}$).

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

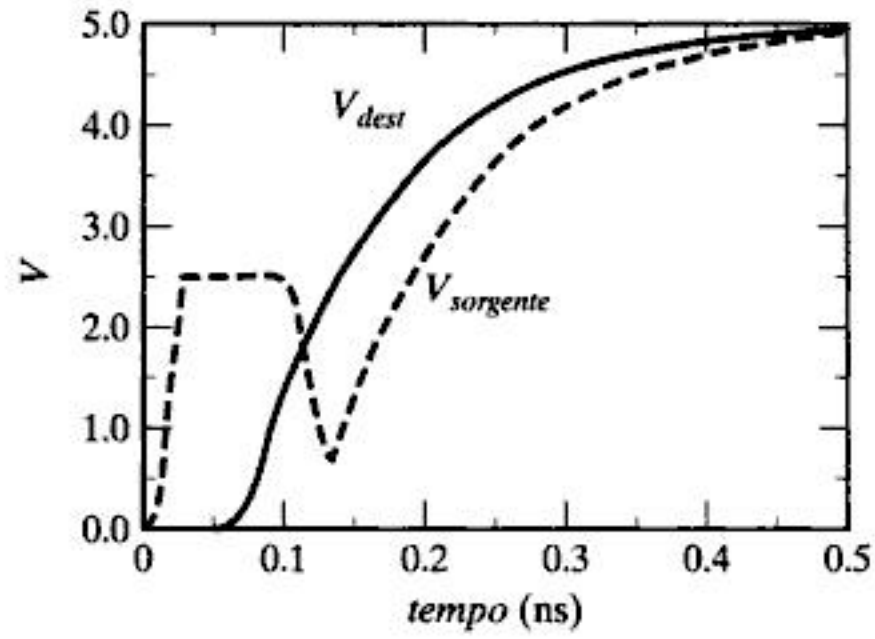


Figura 4-24 Linea di trasmissione terminata capacitivamente: $R_S = 50\ \Omega$, $R_L = \infty$, $C_L = 2$ pF, $Z_0 = 50\ \Omega$, $t_{volo} = 50$ ps.

mente a 0 V, come mostrato nella simulazione. Questo effetto scompare gradualmente, mentre il nodo di uscita converge al suo valore finale.

Il ritardo di propagazione è pari alla somma tra il tempo di volo della linea (50 ps) e il tempo necessario a caricare la capacità ($0.69\ Z_0\ C_L = 69$ ps); ed è esattamente questo ciò che si ottiene dalla simulazione. In generale possiamo dire che sia necessario considerare il carico capacitivo solo quando è comparabile o maggiore della capacità totale della linea di trasmissione [Bakoglu90].

Linee di trasmissione con perdite

Mentre le connessioni su scheda e tra moduli sono sufficientemente spesse e larghe da poter essere trattate come linee di trasmissione senza perdite, per le interconnessioni all'interno di un chip, dove la resistenza della piste è un fattore significativo, le cose stanno in modo diverso. In questo caso è quindi necessario utilizzare il modello delle linee di trasmissione con perdite. Affrontare in modo approfondito le linee con perdita va al di là degli scopi di questo libro, quindi analizzeremo gli effetti della perdita resistiva solamente in modo qualitativo.

La risposta di una *linea RLC* con perdita a un gradino unitario combina la propagazione d'onda con una componente di diffusione. Ciò è mostrato nella Figura 4-25, che riporta la risposta della linea di trasmissione *RLC* in funzione della distanza dalla sorgente. Il gradino si propaga ancora lungo la linea come un'onda, però l'ampiezza di tale onda è attenuata lungo la linea:

$$\frac{V_{step}(x)}{V_{step}(0)} = e^{-\frac{r}{2Z_0}x} \tag{4.35}$$

L'arrivo dell'onda è seguito dalla diminuzione per diffusione del valore a regime al punto x. Più si è lontani dalla sorgente, più la risposta assomiglia al comportamento di una linea *RC* distribuita: di fatto l'effetto resistivo diviene dominante e la linea di trasmissione si

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

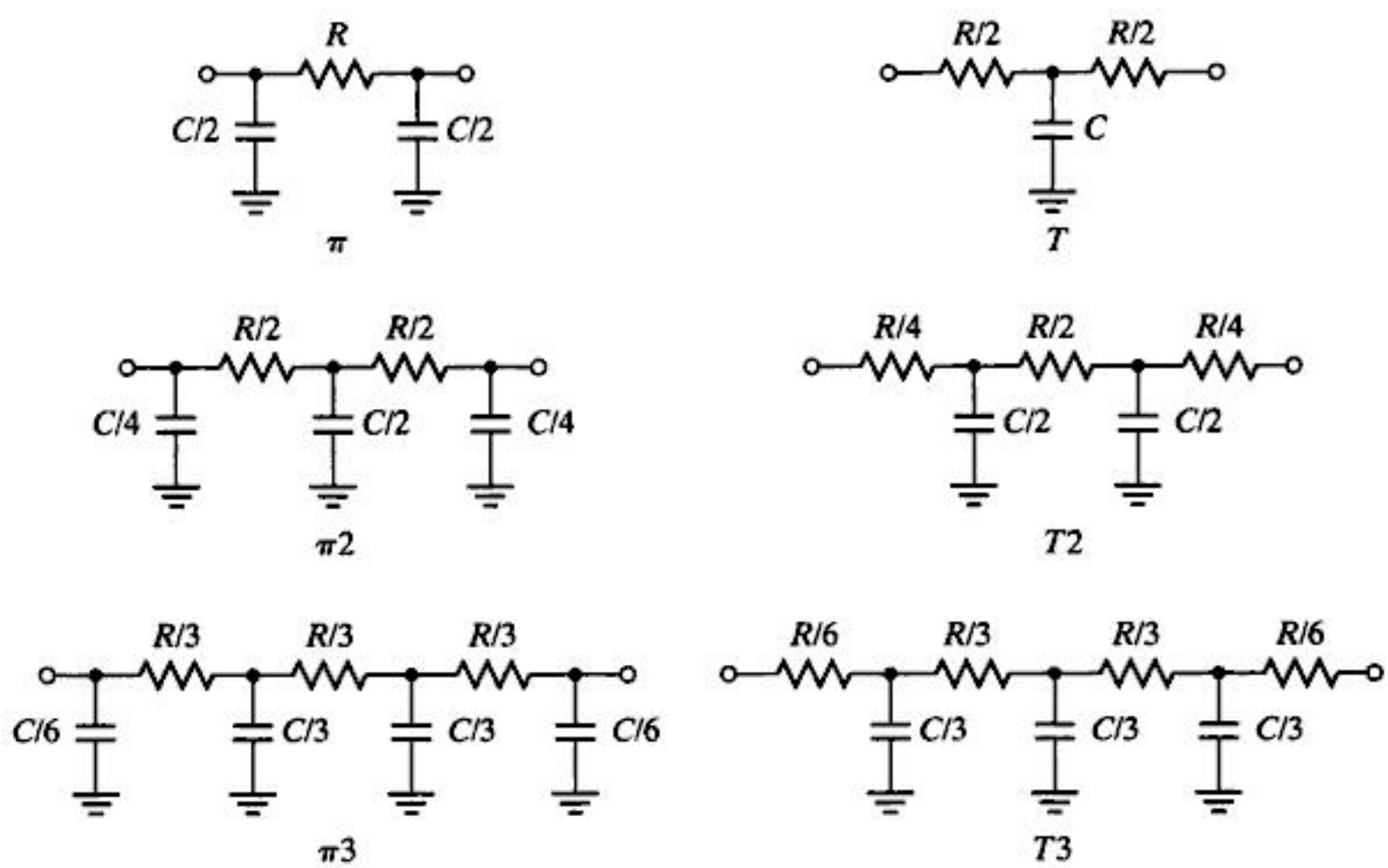


Figura 4-26 Modelli per la simulazione di linee *RC* distribuite.

$$NL = F \cdot TD \tag{4.40}$$

Non sempre è disponibile un modello con perdite. Quando necessario, le perdite possono essere incluse partizionando una linea di trasmissione lunga in sezioni più corte e aggiungendo delle piccole resistenze serie in ciascuna sezione per modellare le perdite della linea. Bisogna comunque porre attenzione nell'utilizzo di questa approssimazione, tenendo presente prima di tutto che la precisione è comunque limitata. In secondo luogo la velocità di simulazione potrebbe essere seriamente rallentata, poiché SPICE sceglie un passo temporale che è minore o al più uguale a metà del valore di *TD*. Per linee di trasmissione corte, tale passo può essere significativamente minore di quanto necessario per l'analisi dei transistor.

4.5.3 Le prospettive: uno sguardo al futuro

Così come abbiamo fatto nel caso del transistor, è interessante analizzare come evolveranno i parametri delle connessioni con la riduzione delle geometrie conseguente all'avanzamento delle tecnologie. Così come si ridurranno le dimensioni dei transistor, sarà necessario ridurre le dimensioni delle connessioni, per sfruttare al meglio i vantaggi derivanti dall'evoluzione dei processi tecnologici.

L'approccio più diretto e semplice consiste nel ridurre le dimensioni delle piste dello stesso fattore *S* di cui vengono ridotte quelle dei transistor (*scaling ideale*). Questo però potrebbe non essere possibile, quantomeno per una delle dimensioni della pista: la sua lunghezza. Si può pure supporre che la lunghezza di *connessioni locali* (piste che connettono gruppi di transistor vicini) scalino nello stesso modo dei transistor. D'altra parte, le *connessioni globali*, che realizzano il collegamento tra moduli estesi e circuiteria di ingresso/uscita, mostreranno un andamento differente (alcuni esempi di questi segnali sono i clock e i

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Riferimenti scientifici

[Bakoglu90] H. Bakoglu, *Circuits, Interconnections and Packaging for VLSI*, Addison-Wesley, 1990.

[Dally98] B. Dally, *Digital Systems Engineering*, Cambridge University Press, 1998.

[Davis98] J. Davis and J. Meindl, "Is Interconnect the Weak Link?" *IEEE Circuits and Systems Magazine*, pp. 30–36, March 1998.

[Elmore48] E. Elmore, "The Transient Response of Damped Linear Networks with Particular Regard to Wideband Amplifiers," *Journal of Applied Physics*, pp. 55–63, January 1948.

[Horowitz83] M. Horowitz, "Timing Models for MOS Circuits," Ph.D. diss., Stanford University, 1983.

[Lin84] T. Lin and C. Mead, "Signal delay in general RC networks," *IEEE Transactions on Computer-Aided Design*, vol. 3, no. 4, pp. 321–349, 1984.

[Rubinstein83] J. Rubinstein, P. Penfield, and M. Horowitz, "Signal Delay in *RC* Networks," *IEEE Transactions on Computer-Aided Design*, vol. CAD-2, pp. 202–211, July 1983.

[Schaper83] L. Schaper and D. Amey, "Improved Electrical Performance Required for Future MOS Packaging," *IEEE Trans. on Components, Hybrids and Manufacturing Technology*, vol. CHMT-6, pp. 282–289, September 1983.

[Sylvester97] D. Sylvester, "High-Frequency VLSI Interconnect Modeling," *Project Report EE241*, UCB, May 1997.

[Tewksbury94] S. Tewksbury, ed., *Microelectronics System Interconnections—Performance and Modeling*, IEEE Press, 1994.

[Vdmeijs84] N. Van De Meijs and J. Fokkema, "VLSI Circuit Reconstruction from Mask Topology," *Integration*, vol. 2, no. 2, pp. 85–119, 1984.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

PARTE

2

Circuito

"Un buon scienziato è una persona dotata di idee originali. Un buon ingegnere è una persona che riesce a fare un progetto funzionante con il minor numero possibile di idee originali"

Freeman Dyson
in "Disturbing the Universe", (1979).

"Un progetto è l'unica cosa che rimane a un progettista quando il tempo e i soldi sono finiti"

James Poole
in "The Fifth 637 Best Things Anybody Ever Said", (1993).

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

5.1 Introduzione

L'invertitore è senz'altro il nucleo di tutti i sistemi digitali e, una volta compreso il suo funzionamento e le sue proprietà, diventa molto più semplice progettare strutture logiche più complesse come porte logiche, sommatori, moltiplicatori e microprocessori. Il comportamento elettrico di questi circuiti complessi può essere ricavato per estrapolazione dai dati ottenuti per l'invertitore. Per esempio, l'analisi dell'invertitore può essere facilmente estesa per spiegare e capire il comportamento di porte logiche non banali (NAND, NOR o XOR), che a loro volta sono alla base di blocchi e moduli funzionali come un moltiplicatore o un processore.

In questo capitolo ci concentreremo su una particolare implementazione dell'invertitore: l'invertitore CMOS statico, il quale è attualmente il più diffuso oggi e merita quindi particolare attenzione. Analizzeremo questa semplice porta logica sulla base delle figure di merito che sono state introdotte nel Capitolo 1:

- *costo*, espresso in termini di complessità e occupazione di area;
- *integrità e robustezza*, ricavata dal comportamento statico;
- *prestazioni*, definite dal comportamento dinamico o transitorio;
- *rendimento energetico*, determinato dal consumo di potenza ed energia.

Usando questa metodologia di analisi svilupperemo un modello della porta logica identificandone i parametri che la caratterizzano. Svilupperemo quindi un criterio per scegliere il valore di questi parametri in modo tale che il risultato sia coerente con le specifiche poste dal progetto. Sebbene molti di questi parametri siano fissati o vincolati dalla tecnologia disponibile, studieremo come essi siano influenzati dallo *scaling e dal'evoluzione delle tecnologie*. Pur concentrandoci, in questo capitolo, solo sull'invertitore CMOS, nel prossimo avremo modo di vedere che le stesse metodologie di studio sono applicabili anche ad altre porte e famiglie logiche.

5.2 L'invertitore CMOS statico: caratteristiche fondamentali

La Figura 5-1 mostra lo schema circuitale di un invertitore CMOS statico. Il suo funzionamento è facilmente intuibile a partire dal modello a interruttore del transistor MOS di cui si è trattato nel Capitolo 3 (Figura 3-26). In prima approssimazione, il transistor può essere rappresentato da un interruttore con una resistenza infinita quando è aperto ($|V_{GS}| < |V_T|$) e una resistenza finita quando è chiuso ($|V_{GS}| > |V_T|$). Ciò si traduce nella seguente interpretazione dell'invertitore: se V_{in} ha un valore logico alto uguale a V_{DD}, il transistor NMOS è attivo e il transistor PMOS è interdetto, ottenendo il circuito mostrato nella Figura 5-2a. In queste condizioni esiste un cammino diretto tra l'uscita e la massa del circuito; a regime l'uscita assume una tensione $V_{out} = 0$ V. Invece, quando la tensione di ingresso ha un valore logico basso (0 V) i transistor NMOS e PMOS sono rispettivamente spento e acceso (Figura 5-2b); l'uscita è connessa attraverso il PMOS all'alimentazione e assume un valore di tensione $V_{out} = V_{DD}$. Questa porta realizza quindi la funzione logica NOT.

Materiale protetto da copyright

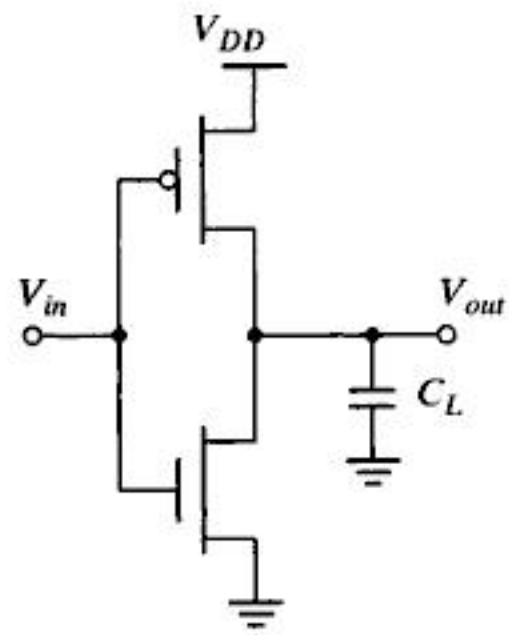


Figura 5-1 Invertitore CMOS statico. V_{DD} è la tensione di alimentazione.

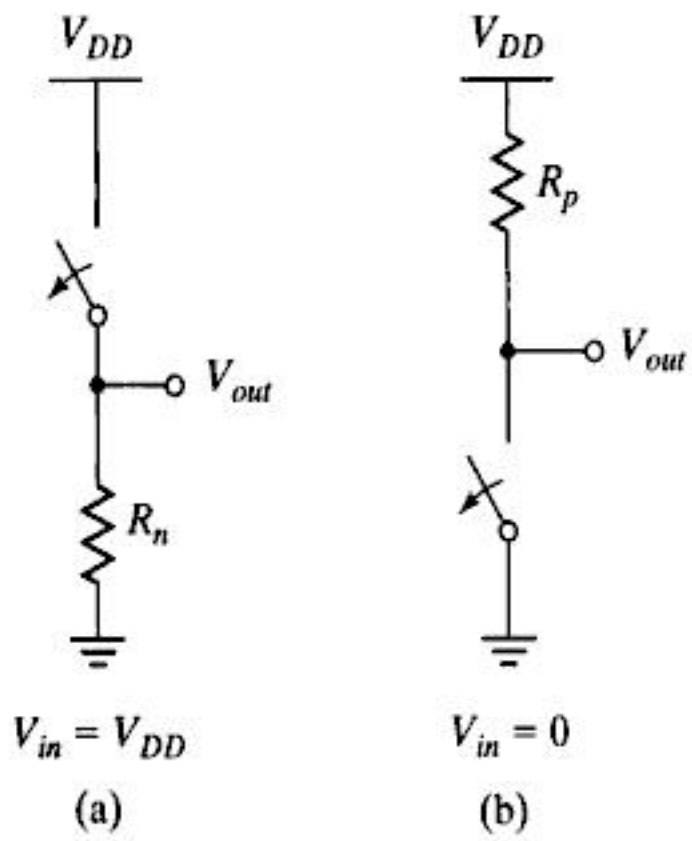


Figura 5-2 Rappresentazione dell'invertitore CMOS mediante interruttori.

Sempre approssimando i transistor come interruttori è possibile ricavare altre importanti proprietà dell'invertitore:

- La tensione di uscita corrispondente ai valori logici alto e basso sono rispettivamente V_{DD} e 0 V; in altre parole l'escursione logica è massima e uguale alla tensione di alimentazione. Ciò si traduce in elevati margini di rumore.
- I livelli logici non dipendono dal rapporto tra le dimensioni dei due transistor, permettendo l'utilizzo di MOS con dimensione minima. Le porte e famiglie logiche che godono di questa proprietà sono dette "*non a rapporto*" (*ratioless logic*), per distinguerle da quelle "*a rapporto*" (*ratioed*), nelle quali i livelli logici sono determinati dal rapporto delle dimensioni dei transistor che la compongono.
- In condizioni stazionarie esiste sempre un cammino diretto con resistenza finita che connette l'uscita o alla massa o all'alimentazione (mai ad entrambe contemporaneamente). Quindi, un invertitore CMOS progettato correttamente ha una *resistenza di*

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

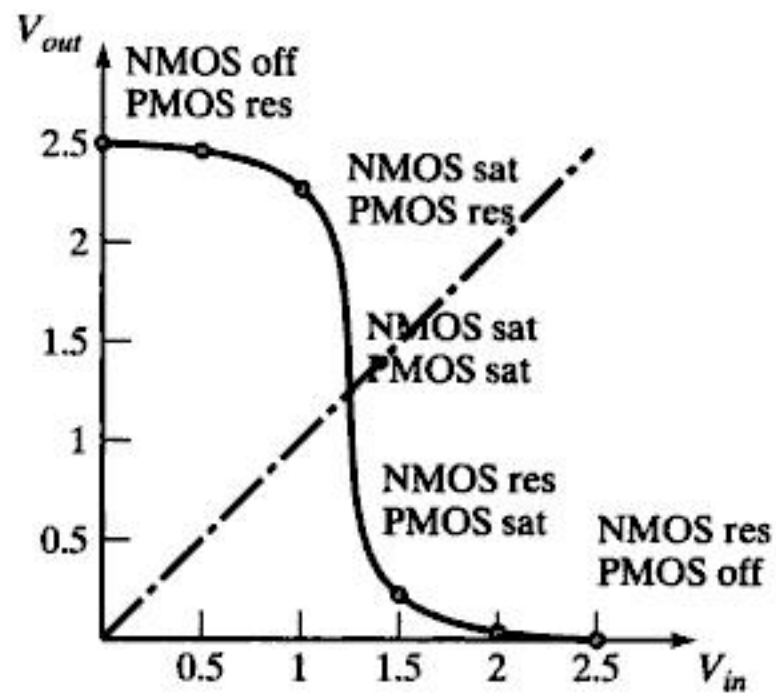


Figura 5-5 VTC di un invertitore CMOS statico derivata graficamente dalle curve in Figura 5-4 (V_{DD} = 2.5 V). Per ciascun punto operativo sono riportate le zone di funzionamento del transistor (off = spento, res = resistiva e sat = saturazione).

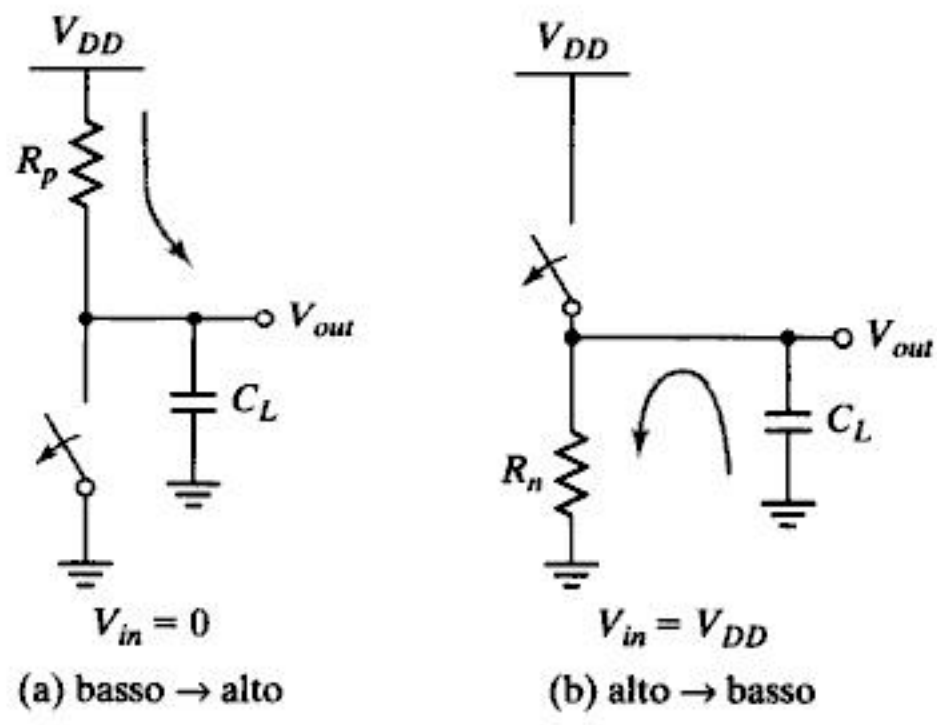


Figura 5-6 Analisi della risposta dinamica dell'invertitore CMOS con il modello ad interruttore del transistor.

Prima di entrare nei dettagli del funzionamento dell'invertitore CMOS, è opportuno fare un'analisi qualitativa del comportamento dinamico della porta logica. La risposta dinamica è dominata principalmente dalla capacità all'uscita, C_L, che è composta dalle capacità parassite dei drain dei due transistor, dalla capacità delle piste di interconnessione e dalle capacità di ingresso delle porte logiche che l'invertitore sta pilotando (fan-out). Supponendo per ora che la tensione di ingresso commuti istantaneamente, possiamo avere un'idea della risposta transitoria utilizzando ancora una volta il modello a interruttore. Consideriamo una transizione basso-alto (vedi Figura 5-6a): la risposta dinamica è determinata dal tempo che la porta impiega a carica la capacità C_L attraverso la resistenza R_p. Nell'Esempio 4.5, abbiamo visto che il tempo di propagazione di un circuito di questo tipo è proporzionale alla costante di tempo R_pC_L. **Quindi, una porta logica veloce è costruita mantenendo bassa la capacità del nodo di uscita e/o diminuendo la resi-**

Materiale protetto da copyright

stenza del transistor. La seconda condizione è ottenuta aumentando il fattore di forma W/L del transistor. Considerazioni simili sono valide per la transizione alto-basso (Figura 5-6b), che è determinata dalla costante di tempo R_nC_L. Occorre tener presente che la resistenza dei transistor NMOS e PMOS non è costante, ma dipende in modo non lineare dalla tensione ai capi del dispositivo. Ciò complica notevolmente il calcolo del tempo di propagazione (l'analisi approfondita del comportamento dinamico dell'invertitore sarà fatta nel paragrafo 5.4).

5.3 Robustezza dell'invertitore CMOS: il comportamento statico

Nella precedente analisi qualitativa è stata ricavata la forma della VTC dell'invertitore CMOS e i livelli logici della tensione di uscita V_{OH} e V_{OL}; rispettivamente V_{DD} e 0 V. Per completare l'analisi del comportamento statico restano da determinare i valori V_M, V_{IH}, V_{IL} e i margini di rumore.

5.3.1 Soglia di commutazione

La soglia di commutazione (o tensione di soglia logica) V_M è definita come il punto della VTC in cui la tensione di uscita uguaglia la tensione di ingresso e può essere ricavato graficamente dall'intersezione della VTC con la retta $V_{in} = V_{out}$ (vedi Figura 5-5). In questo punto entrambi i transistor sono in zona di saturazione poiché $V_{DS} = V_{GS}$. L'espressione analitica di V_M si può ottenere imponendo l'uguaglianza delle correnti dei due transistor e risolvendo l'equazione nell'ipotesi che la tensione di alimentazione sia sufficientemente alta in modo che entrambi i MOS siano in saturazione di velocità ($V_{DSAT} < V_M - V_T$). Trascurando per semplicità l'effetto della modulazione della lunghezza di canale, otteniamo:

$$k_nV_{DSATn}\left(V_M - V_{Tn} - \frac{V_{DSATn}}{2}\right) + k_pV_{DSATp}\left(V_M - V_{DD} - V_{Tp} - \frac{V_{DSATp}}{2}\right) = 0 \qquad (5.2)$$

e risolvendo nella variabile V_M, risulta:

$$V_M = \frac{\left(V_{Tn} + \frac{V_{DSATn}}{2}\right) + r\left(V_{DD} + V_{Tp} + \frac{V_{DSATp}}{2}\right)}{1 + r} \quad \text{con} \quad r = \frac{k_pV_{DSATp}}{k_nV_{DSATn}} = \frac{\upsilon_{satp}W_p}{\upsilon_{satn}W_n} \qquad (5.3)$$

l'ultima semplificazione nell'espressione di r è stata fatta assumendo uguale spessore dell'ossido di gate per NMOS e PMOS. Se V_{DD} è molto grande rispetto alla tensione di soglia e alla tensione di saturazione, l'Eq. (5.3) può essere semplificata ulteriormente:

$$V_M \approx \frac{rV_{DD}}{1 + r} \qquad (5.4)$$

L'Eq. (5.4) stabilisce che la tensione di soglia logica è determinata dal rapporto, r, tra la "forza" conduttiva del PMOS e quella del transistor NMOS. Come già osservato, generalmente si desidera che il valore di V_M sia vicino al punto medio dell'escursione ($V_{DD}/2$), poiché

Materiale protetto da copyright

in questo modo si ottiene una VTC simmetrica e margini di rumore a livello alto e basso paragonabili. Nelle ipotesi dell'Eq. (5.4) la simmetria della VTC richiede un valore di r all'incirca unitario, che equivale a dimensionare il PMOS con fattore di forma $(W/L)_p = (W/L)_n \times (V_{DSATn}k'_n)/(V_{DSATn}k'_p)$. Per aumentare il valore di V_M, è necessario aumentare le dimensioni del PMOS oltre questo valore, mentre per diminuire la soglia di commutazione è necessario realizzare il transistor NMOS più grande.

Dall'Eq. (5.2) possiamo calcolare il rapporto tra i fattori di forma dei due transistor in modo tale da ottenere il valore V_M desiderato per la soglia logica:

$$\frac{(W/L)_p}{(W/L)_n} = \frac{k'_n V_{DSATn}(V_M - V_{Tn} - V_{DSATn}/2)}{k'_p V_{DSATp}(V_{DD} - V_M + V_{Tp} + V_{DSATp}/2)} \tag{5.5}$$

Quando si utilizza questa espressione bisogna accertarsi che le ipotesi fatte sulla zona di funzionamento dei transistor siano verificate.

Problema 5.1 Soglia di commutazione dell'invertitore con MOS a canale lungo o per bassi valori della tensione di alimentazione

L'espressione precedente è stata calcolata nell'ipotesi che entrambi i transistor lavorino in saturazione di velocità. Quando si ha a che fare con dispositivi a canale lungo o quando la tensione di alimentazione ($V_M - V_T < V_{DSAT}$) è sufficientemente bassa, la velocità dei portatori nel canale non satura. In questo caso si ottiene la seguente espressione per la soglia logica:

$$V_M = \frac{V_{Tn} + r(V_{DD} + V_{Tp})}{1 + r} \quad \text{with } r = \sqrt{\frac{-k_p}{k_n}} \tag{5.6}$$

Dimostrare questa relazione.

Tecniche di progettazione: ottimizzare i margini di rumore

Nella progettazione di circuiti CMOS statici, per i quali siano richiesti alti margini di immunità al rumore, è consigliabile bilanciare le dimensioni dei due transistor realizzando il PMOS più largo del transistor NMOS, ottenendo così una VTC simmetrica. Il valore del rapporto è dato dall'Eq. (5.5). ■

Esempio 5.1 Soglia logica di un invertitore CMOS

Calcoliamo le dimensioni dei transistor PMOS e NMOS in modo tale che la soglia di commutazione dell'invertitore CMOS, implementato nella nostra tecnologia CMOS di riferimento da 0.25 μm, sia posizionata al centro dell'escursione logica. Usiamo i valori dei parametri presentati nell'Esempio 3.7; assumiamo che il circuito sia alimentato a 2.5 V e che un dispositivo con dimensioni minime abbia un fattore di forma di 1.5.

Usando l'Eq. (5.5), otteniamo:

Materiale protetto da copyright

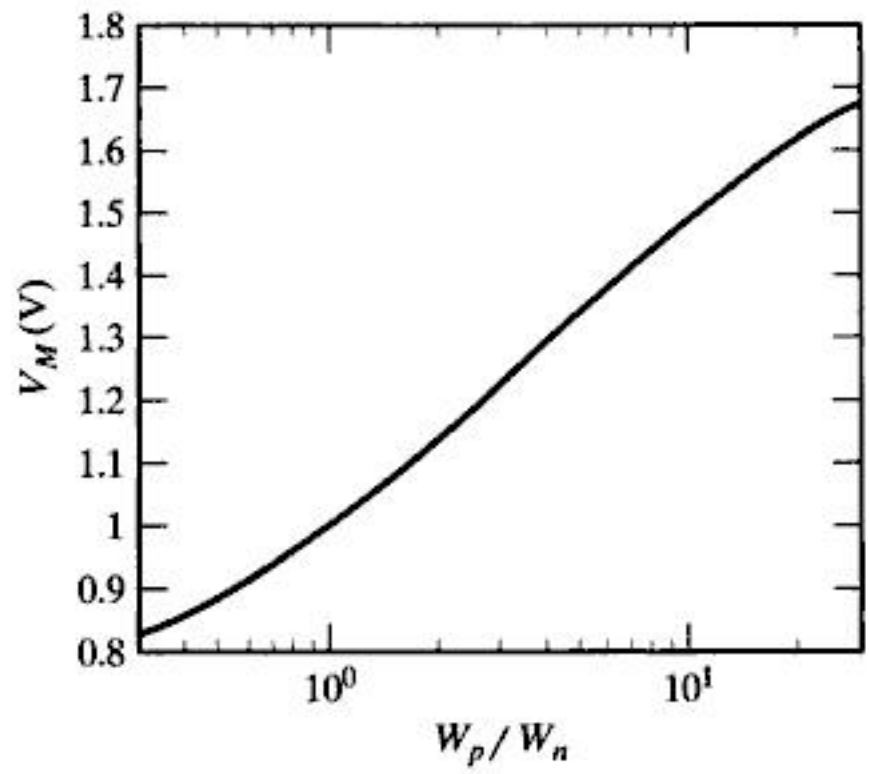


Figura 5-7 Soglia di commutazione simulata di un invertitore CMOS in funzione del rapporto tra le dimensioni dei transistor PMOS e NMOS (tecnologia CMOS da 0.25-μm, V_{DD} = 2.5 V).

$$\frac{(W/L)_p}{(W/L)_n} = \frac{115 \times 10^{-6}}{30 \times 10^{-6}} \times \frac{0{,}63}{1{,}0} \times \frac{(1{,}25 - 0{,}43 - 0{,}63/2)}{(1{,}25 - 0{,}4 - 1{,}0/2)} = 3{,}5$$

Il grafico della Figura 5-7 rappresenta la soglia di commutazione in funzione del rapporto tra le larghezze W dei dispositivi PMOS e NMOS ottenuta da una simulazione circuitale. Dalla simulazione osserviamo che per ottenere V_M = 1.25 V il rapporto tra W_p e W_n deve essere 3.4, confermando quanto ottenuto dall' Eq. (5.5).

Un'occhiata al grafico in Figura 5-7 porta ad alcune considerazioni:

1. V_M è relativamente poco sensibile a una variazione delle dimensioni dei dispositivi: piccole variazioni del rapporto W_p/W_n (per esempio da 3 a 2.5) non ha grandi effetti nella caratteristica di trasferimento. È consuetudine abbastanza diffusa a livello industriale realizzare il PMOS leggermente inferiore a quello che fornirebbe un esatta simmetria della VTC. Nell'esempio precedente, scegliendo W_p/W_n = 3, 2.5 e 2, si ottiene una soglia di commutazione V_M = 1.22 V, 1.18 V e 1.13 V, rispettivamente.
2. Cambiare il rapporto W_p/W_n ha come effetto lo spostamento della regione di transizione della VTC. Aumentare la larghezza del transistor PMOS (o NMOS) sposta V_M verso V_{DD} (o la massa). Questa caratteristica è molto importante dato che in alcune applicazioni è richiesta una VTC asimmetrica, come dimostra l'esempio della Figura 5-8: il segnale V_{in} è caratterizzato da un livello basso molto rumoroso; se questo segnale passasse attraverso un invertitore con VTC simmetrica, all'uscita potrebbero aversi delle commutazioni indesiderate (Figura 5-8a). Il problema può essere risolto aumentando la soglia di commutazione dell'invertitore al di sopra del livello del rumore, ottenendo la risposta illustrata nella Figura 5-8b. Più avanti

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

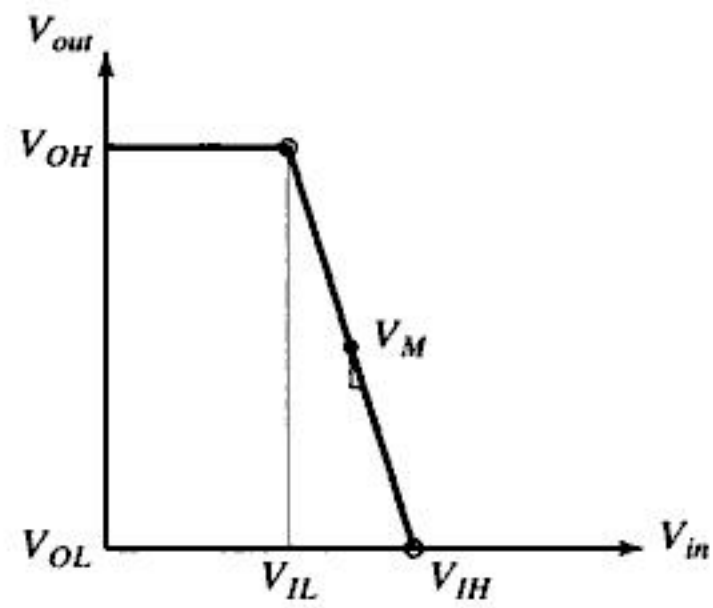


Figura 5-9 Approssimazione della VTC dell'invertitore con una funzione lineare a tratti per il calcolo di V_{IL} e V_{IH}.

prima stima del progetto. L'approccio appena descritto porta alle seguenti espressioni per l'ampiezza della regione di transizione $V_{IH} - V_{IL}$, i punti V_{IH} e V_{IL} e i margini di rumore NM_H e NM_L:

$$
\begin{aligned}
V_{IH} - V_{IL} &= -\frac{(V_{OH} - V_{OL})}{g} = \frac{-V_{DD}}{g} \\
V_{IH} &= V_M - \frac{V_M}{g} \qquad V_{IL} = V_M + \frac{V_{DD} - V_M}{g} \\
NM_H &= V_{DD} - V_{IH} \qquad NM_L = V_{IL}
\end{aligned}
\tag{5.7}
$$

Queste espressioni giustificano la necessità di un elevato guadagno nella regione di transizione. Se il idealmente il guadagno fosse infinito, le espressioni dei margini di rumore si semplificherebbero in $NM_H = V_{OH} - V_M$ e $NM_L = V_M - V_{OL}$, coprendo l'intera escursione logica.

Resta ora da calcolare il guadagno g dell'invertitore CMOS. Ancora una volta, ipotizziamo che entrambi i MOS siano in condizioni di saturazione di velocità. Appare evidente dalla Figura 5-4 che il guadagno della VTC dipende fortemente dalla pendenza della corrente di saturazione dei due transistor: non è quindi possibile trascurare l'effetto della modulazione della lunghezza di canale, altrimenti si otterrebbe un valore infinito di guadagno. Il guadagno può essere calcolato a partire dalla seguente equazione, valida in un intorno di V_M:

$$
\begin{aligned}
&k_n V_{DSATn}\left(V_{in} - V_{Tn} - \frac{V_{DSATn}}{2}\right)(1 + \lambda_n V_{out}) + \\
&\qquad k_p V_{DSATp}\left(V_{in} - V_{DD} - V_{Tp} - \frac{V_{DSATp}}{2}\right)(1 + \lambda_p V_{out} - \lambda_p V_{DD}) = 0
\end{aligned}
\tag{5.8}
$$

Derivando l'Eq. (5.8) rispetto a V_{in} e risolvendo l'equazione nella variabile dV_{out}/dV_{in}, si ottiene:

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

5.3.3 Robustezza e tecnologia

Dispersione delle caratteristiche dei dispositivi

Sebbene il progetto di una porta logica sia fatto considerando le condizioni operative nominali e i valori tipici dei parametri dei dispositivi, occorre tenere presente che la temperatura di funzionamento può variare in un certo intervallo e che i parametri e le caratteristiche dei dispositivi fabbricati possono deviare dal valore nominale usato in fase di progettazione e ottimizzazione. Fortunatamente, le caratteristiche statiche dell'invertitore CMOS si dimostrano abbastanza insensibili a tali variazioni, permettendo al circuito di rimanere perfettamente funzionante in un ampio spettro di condizioni operative. Ciò appare evidente nella Figura 5-7, che mostra come le variazioni delle dimensioni dei dispositivi abbiano ripercussioni modeste nel valore della tensione di commutazione dell'invertitore. Per dimostrare la robustezza dell'invertitore abbiamo simulato la caratteristica di trasferimento sostituendo i valori nominali dei parametri del dispositivo con i loro valori massimi e minimi dovuti alle incertezze del processo di fabbricazione. Nella Figura 5-11 sono riportati due casi limite: un NMOS migliore di quello nominale (veloce) combinato con un PMOS peggiore (lento), e viceversa. Confrontando le due curve risultanti con la VTC nominale si vede che la funzionalità del circuito non risulta per niente compromessa, e che queste variazioni provocano, al massimo, una modesta traslazione della tensione di commutazione. Questa robustezza, che assicura la funzionalità della porta in un ampio spettro di condizioni operative, è uno dei fattori che hanno largamente contribuito alla popolarità della logica CMOS statica.

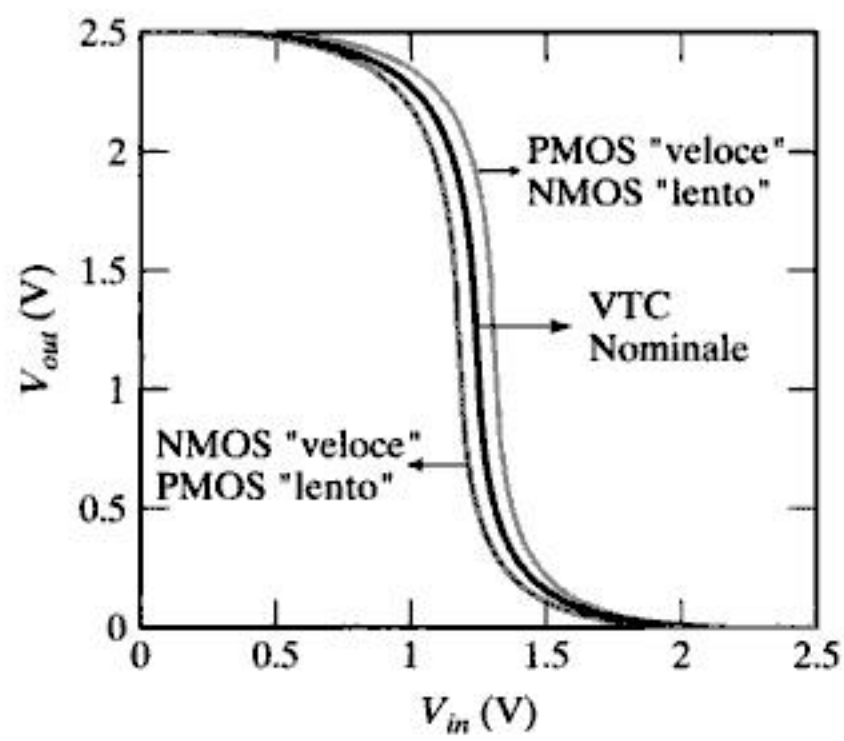


Figura 5-11 Conseguenze della dispersione dei parametri di processo sulla VTC di un invertitore CMOS statico. Il dispositivo "veloce" ha: ossido di gate più sottile (– 3 nm), minore lunghezza di canale (– 25 nm), maggiore larghezza di canale (+ 30 nm) e tensione di soglia inferiore (– 60 mV). Un dispositivo "lento" è caratterizzato da variazioni di segno opposto.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

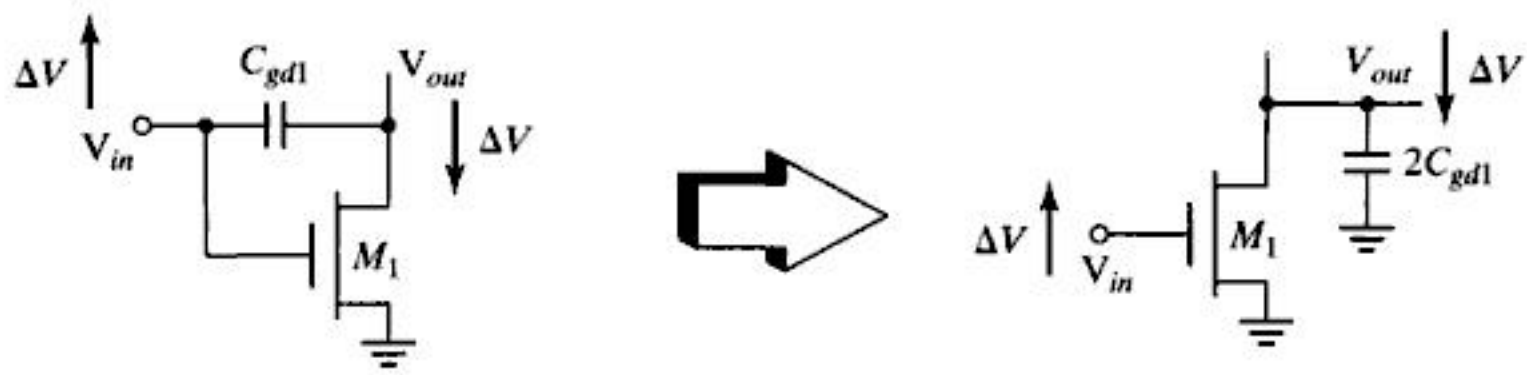


Figura 5-14 Effetto Miller: una capacità che ha una variazione di tensione identica ma di segno opposto a entrambi i terminali può essere sostituita da una capacità riferita a massa il cui valore è doppio rispetto a quello originale.

un ruolo determinante, poiché è quasi completamente localizzata tra gate e substrato (se spento) o tra gate e source (se saturo), come discusso nel Capitolo 3.

Questo modello semplificato, concentrando tutte le capacità in un unico contributo, richiede che anche queste due capacità flottanti tra gate e drain siano sostituite da un'unica capacità riferita a massa. Per fare ciò è necessario tenere conto dell'effetto Miller: durante la commutazione basso-alto o alto basso, i terminali della capacità gate-drain hanno potenziali che variano contemporaneamente e in direzione opposta (vedi Figura 5-14). La variazione di tensione effettiva ai capi della capacità è quindi doppia rispetto alla variazione di tensione del nodo di uscita. Per tenere conto dell'effetto Miller, è necessario che la capacità equivalente verso massa abbia un valore doppio rispetto alla capacità flottante tra gate e drain.

Usiamo quindi la seguente espressione per la capacità gate-drain: $C_{gd} = 2\ C_{GD0}W$ (con C_{GD0} la capacità di sovrapposizione per unità di larghezza W, in analogia al modello SPICE). Per maggiori dettagli sull'effetto Miller consultare libri di testo come [Sedra87, p. 57].[2]

Capacità della diffusione di drain: C_{db1} e C_{db2}

La capacità della diffusione di drain è dovuta alla giunzione *pn* tra drain e substrato polarizzata inversamente. Purtroppo questa capacità è fortemente non lineare e dipende dalla polarizzazione applicata. Nel Capitolo 3 abbiamo appreso che il modo migliore per semplificare l'analisi è sostituire una capacità non lineare con una lineare ad essa equivalente nell'escursione di tensione di interesse. Per mettere in relazione la capacità equivalente linearizzata e la capacità di giunzione con polarizzazione nulla applicata abbiamo introdotto il fattore moltiplicativo K_{eq}:

$$C_{eq} = K_{eq}C_{j0} \tag{5.13}$$

dove C_{j0} è la capacità di giunzione per unità di area con polarizzazione nulla. Per comodità riprendiamo l'Eq. (3.11) scritta nel modo seguente:

$$K_{eq} = \frac{-\phi_0^m}{(V_{high} - V_{low})(1-m)}[(\phi_0 - V_{high})^{1-m} - (\phi_0 - V_{low})^{1-m}] \tag{5.14}$$

[2]L'effetto Miller qui discusso può essere visto come un caso particolare e semplificato di quanto accade nei comuni amplificatori analogici. In un invertitore il guadagno a grandi segnali tra ingresso e uscita è uguale a −1.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Grazie a questa tecnica la capacità di giunzione può essere sostituita con una capacità lineare equivalente e trattata come ogni altro contributo. L'approssimazione lineare introduce un errore modesto nelle forme d'onda della tensione e della corrente, e ha un effetto trascurabile nel calcolo del tempo di ritardo.

Capacità dell'interconnessione: C_w

Il contributo capacitivo introdotto dalle piste di interconnessione dipende dalla lunghezza e dalla larghezza delle piste stesse, ed è funzione della distanza dall'uscita dell'invertitore al carico e dal numero di porte connesse all'uscita (fan-out). Come abbiamo visto nel Capitolo 4, questo contributo è di crescente importanza con l'evoluzione e lo scaling della tecnologia.

Capacità di ingresso dello stadio successivo: C_{g3} e C_{g4}

Supponiamo che l'invertitore stia pilotando un altro invertitore uguale a se stesso (M_3 e M_4). Quindi alla capacità del nodo di uscita dobbiamo aggiungere il carico proveniente dalle capacità dei gate dei transistor M_3 e M_4:

$$\begin{aligned} C_{fan\text{-}out} &= C_{gate}(\text{NMOS}) + C_{gate}(\text{PMOS}) \\ &= (C_{GSOn} + C_{GDOn} + W_n L_n C_{ox}) + (C_{GSOp} + C_{GDOp} + W_p L_p C_{ox}) \end{aligned} \tag{5.15}$$

Questa espressione esemplifica la situazione reale per due motivi:

- L'Eq. (5.15) assume che tutti i contributi capacitivi siano connessi tra V_{out} e la massa (o V_{DD}), ignorando l'effetto Miller sulle capacità gate–drain. Ciò ha un effetto modesto sull'accuratezza del risultato, dato che possiamo assumere con buona approssimazione che le porte connesse in uscita non commutano prima che il nodo V_{out} raggiunga il 50% della sua escursione: V_{out2} rimane quindi costante durante l'intervallo di interesse.
- Una seconda approssimazione riguarda la capacità di canale del secondo invertitore, assunta come costante su tutto l'intervallo di tensione. Ciò non è completamente vero, come abbiamo avuto modo di vedere nel Capitolo 3: la capacità di canale è funzione della regione di funzionamento e varia da circa $(2/3)WLC_{ox}$ (in saturazione) a WLC_{ox} (zona lineare e di interdizione). Un brusco calo della capacità totale di ingresso dell'invertitore avviene anche quando uno dei due transistor si accende, come evidenzia la Figura 3-31. Durante la prima metà dell'escursione possiamo assumere che uno dei transistor sia sempre in zona lineare, mentre l'altro sia inizialmente spento e poi si accenda in saturazione. Trascurando le variazioni nel valore della capacità di canale e considerando che uno dei transistor sia sempre in saturazione e l'altro in zona lineare, otteniamo una stima pessimistica con un errore di circa il 10%, che è sicuramente accettabile per un'analisi al primo ordine.

Esempio 5.4 Capacità di un invertitore CMOS in tecnologia da 0.25 μm

Il layout di un invertitore CMOS simmetrico e con dimensione minima realizzato con una tecnologia da 0.25 μm è disegnato nella Figura 5-15. La tensione di alimentazione

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

del drain del transistor PMOS può essere ricavato in modo simile e risulta: AD = 5 × 9 λ^2 = 45 λ^2, or 0.7 μm^2; PD = 5 + 9 + 5 = 19 λ, or 2.375 μm (la forma rettangolare della diffusione rende il conto più semplice.

Queste informazioni sulle dimensioni fisiche combinate insieme alle approssimazioni date in precedenza per i vari contributi capacitivi ci forniscono il valore finale di C_L. I parametri relativi alle capacità dei transistor NMOS e PMOS per l'ipotetica tecnologia di riferimento che abbiamo adottato sono stati forniti nella Tabella 3-5 e sono riportati qui sotto per comodità:

capacità di sovrapposizione: CGD0(NMOS) = 0.31 fF/μm; CGDO(PMOS) = 0.27 fF/μm
capacità bottom plate: CJ(NMOS) = 2 fF/μm^2; CJ(PMOS) = 1.9 fF/μm^2
capacità della sidewall: CJSW(NMOS) = 0.28 fF/μm; CJSW(PMOS) = 0.22 fF/μm
capacità dell'ossido di gate: C_{ox}(NMOS) = C_{ox}(PMOS) = 6 fF/μm^2

Infine, dobbiamo considerare anche il contributo capacitivo proveniente dalla pista di interconnessione tra le due porte logiche implementate parte in metal-1 e parte in polisilicio. Generalmente gli strumenti automatici di estrazione parametrica dal layout forniscono stime molto accurate delle capacità parassite. Tuttavia uno sguardo al layout ci aiuta a dare una prima stima di tale capacità. L'area delle piste di metal-1 e di polisilicio che non sono sovrapposte alle aree attive dei transistor, sono uguali rispettivamente a 42 λ^2 e 72 λ^2. Usando i parametri parassiti delle interconnessioni riportati nella Tabella 4-2, otteniamo la capacità totale dell'interconnessione (attenzione: in questo semplice esempio abbiamo trascurato i contributi capacitivi del bordo delle piste; effettivamente,

Tabella 5-2 Contributi a C_L (per una commutazione alto-basso e basso-alto).

Capacità	Espressione	Valore (fF) (alto → basso)	Valore (fF) (basso → alto)
C_{gd1}	$2\ CGD0_n\ W_n$	0.23	0.23
C_{gd2}	$2\ CGD0_p\ W_p$	0.61	0.61
C_{db1}	$K_{eqn}\ AD_n\ CJ + K_{eqswn}\ PD_n\ CJSW$	0.66	0.90
C_{db2}	$K_{eqp}\ AD_p\ CJ + K_{eqswp}\ PD_p\ CJSW$	1.5	1.15
C_{g3}	$(CGD0_n+CGSO_n)\ W_n + C_{ox}\ W_n\ L_n$	0.76	0.76
C_{g4}	$(CGD0_p+CGSO_p)\ W_p + C_{ox}\ W_p\ L_p$	2.28	2.28
C_w	*Estratta numericamente*	0.12	0.12
C_L	Σ	6.1	6.0

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

fatta sottoposta a un gradino di tensione, è proporzionale alla costante di tempo della rete costituita dalla resistenza di pull-down e dalla capacità di carico:

$$t_{pHL} = \ln(2)R_{eqn}C_L = 0{,}69R_{eqn}C_L \tag{5.18}$$

Analogamente, possiamo calcolare il tempo di ritardo per una transizione basso-alto:

$$t_{pLH} = 0{,}69R_{eqp}C_L \tag{5.19}$$

dove R_{eqp} è la resistenza equivalente del transistor PMOS sull'intervallo di interesse. In questi calcoli abbiamo assunto che la capacità di carico abbia lo stesso valore sia nella commutazione basso-alto sia in quella alto-basso. È possibile dimostrare che questa approssimazione è abbastanza buona nel caso dell'esempio precedente. Il tempo di ritardo complessivo è definito come la media dei due tempi appena calcolati:

$$t_p = \frac{t_{pHL} + t_{pLH}}{2} = 0{,}69C_L\left(\frac{R_{eqn} + R_{eqp}}{2}\right) \tag{5.20}$$

Molto spesso si desidera che i tempi di propagazione di una porta logica siano gli stessi per entrambe le commutazioni del segnale di ingresso. A tale scopo è necessario realizzare i due transistor in modo tale che abbiano approssimativamente lo stesso valore di resistenza equivalente.

Esempio 5.5 Tempo di propagazione di un invertitore CMOS statico

Per calcolare il tempo di propagazione dell'invertitore illustrato nella Figura 5-15, facciamo uso dell'Eq. (5.18) e dell'Eq. (5.19). La capacità di carico C_L è già stata calcolata nell'Esempio 5.4, mentre le resistenze equivalenti dei transistor, nella nostra tecnologia di riferimento da 0.25 μm, sono riportate nella Tabella 3-3. A una tensione di alimentazione di 2.5 V le resistenze dei transistor NMOS e PMOS con fattore di forma unitario ($W/L = 1$) sono rispettivamente 13 kΩ e 31 kΩ. Dal layout dell'invertitore calcoliamo che i fattori di forma dei transistor sono 1.5 (NMOS) e 4.5 (PMOS). Per semplicità supponiamo che le differenze tra le dimensioni disegnate e quelle effettive siano abbastanza piccole da essere trascurabili. Otteniamo i seguenti valori per i tempi di propagazione:

$$t_{pHL} = 0{,}69 \times \left(\frac{13\ \text{k}\Omega}{1{,}5}\right) \times 6{,}1\ \text{fF} = 36\ \text{ps}$$

$$t_{pLH} = 0{,}69 \times \left(\frac{31\ \text{k}\Omega}{4{,}5}\right) \times 6{,}0\ \text{fF} = 29\ \text{ps}$$

e

$$t_p = \left(\frac{36 + 29}{2}\right) = 32{,}5\ \text{ps}$$

Materiale protetto da copyright

L'accuratezza di questi calcoli può essere valutata eseguendo una simulazione con SPICE del circuito estratto dal layout della Figura 5-15. La risposta dinamica simulata del circuito è riportata nel grafico della Figura 5-16, e ci fornisce i due tempi di propagazione pari a 39.9 ps e 31.7 ps per le commutazioni alto-basso e basso-alto rispettivamente. L'analisi manuale si rivela abbastanza precisa, tenuto conto di tutte le approssimazioni che sono state fatte per arrivare alla formula finale. Notiamo in particolare che la tensione di uscita, simulata nella Figura 5-16, ha un picco superiore a V_{DD} in corrispondenza della commutazione del segnale d'ingresso. Ciò è causato dalla capacità parassita tra gate e drain dei transistor che compongono l'invertitore, che produce un accoppiamento capacitivo tra il nodo di ingresso e quello di uscita: la veloce commutazione dell'ingresso è parzialmente trasferita all'uscita grazie a questa capacità causando l'aumento di tensione prima che i transistor rispondano al cambiamento di tensione all'ingresso. Questo aumento iniziale della tensione di uscita ha un impatto negativo sulle prestazioni dell'invertitore e spiega perché i tempi di propagazione simulati sono maggiori di quelli stimati tramite il calcolo manuale.

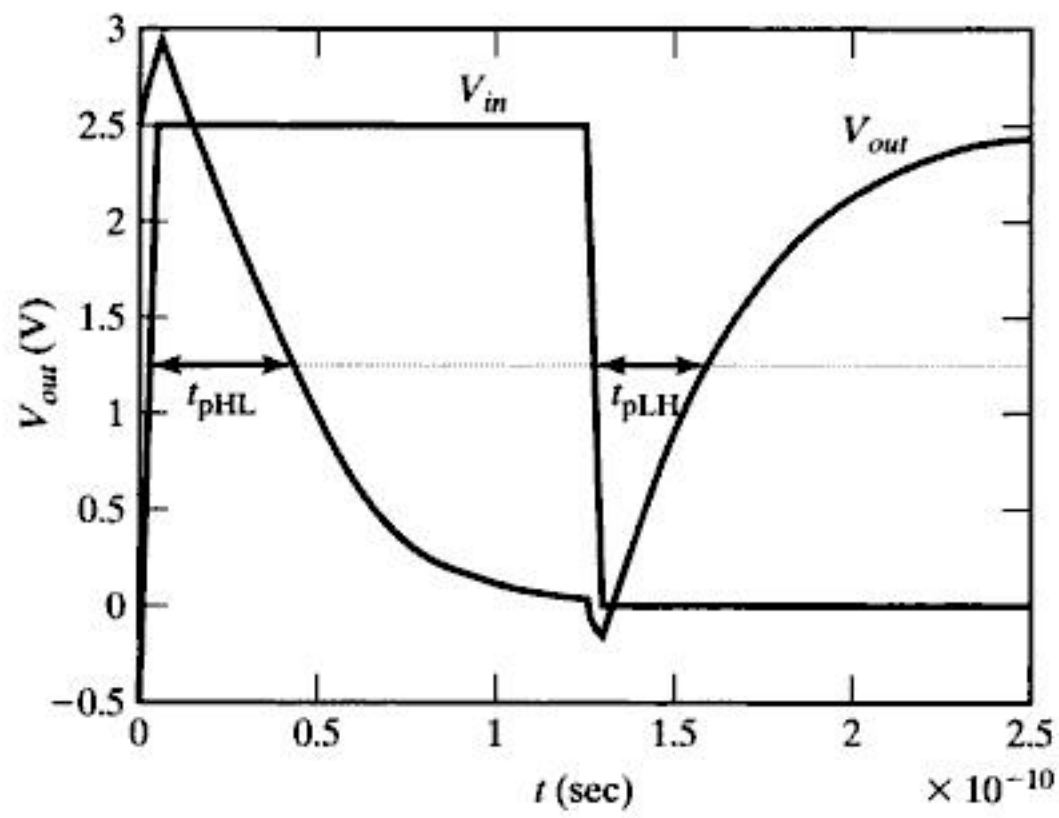


Figura 5-16 Simulazione della risposta dinamica dell'invertitore della Figura 5-15.

ATTENZIONE: Quest'esempio potrebbe dare l'impressione che l'analisi manuale sia sempre in grado di fornire risultati approssimati molto vicini alla realtà, ma ciò non è sempre vero. A volte vi possono essere notevoli discrepanze tra l'analisi semplificata al primo ordine e i risultati di elaborate simulazioni che tengano conto di tutti i fattori del caso, utilizzando modelli più accurati e complessi. Lo scopo dell'analisi manuale è di fornire una visione semplice e intuitiva del circuito e evidenziare quelli che sono i parametri dominanti. Al contrario, quando è richiesta una stima qualitativa è indispensabile ricorrere alla simulazione. Il risultato dell'esempio precedente deve essere considerato come un colpo di fortuna.

Materiale protetto da copyright

La domanda più ovvia che si pone ora un progettista riguarda il modo in cui ottimizzare il tempo di propagazione di una porta logica (l'invertitore, in questo caso). Per trovare una risposta a questa domanda è necessario rendere espliciti i parametri che determinano direttamente il ritardo, espandendo l'espressione di R_{eq} nell'Eq. (5.20) del tempo di propagazione. Mettendo insieme l'Eq. (5.18) e l'Eq. (5.17), e trascurando il fattore di modulazione della lunghezza di calane λ, otteniamo la seguente espressione per t_{pHL} (analogamente si può procedere per t_{pLH}):

$$t_{pHL} = 0{,}69\frac{3}{4}\frac{C_L V_{DD}}{I_{DSATn}} = 0{,}52\frac{C_L V_{DD}}{(W/L)_n k'_n V_{DSATn}(V_{DD} - V_{Tn} - V_{DSATn}/2)} \tag{5.21}$$

Nella maggior parte dei casi la tensione di alimentazione è alta abbastanza da poter scrivere $V_{DD} >> V_{Tn} + V_{DSATn}/2$. In queste ipotesi il tempo di ritardo diviene praticamente indipendente dalla tensione di alimentazione:

$$t_{pHL} \approx 0{,}52\frac{C_L}{(W/L)_n k'_n V_{DSATn}} \tag{5.22}$$

Osserviamo che questa espressione è un risultato di un analisi approssimata al primo ordine e che aumentando la tensione di alimentazione otteniamo un sensibile, sebbene modesto, miglioramento delle prestazioni, dovuto al fattore di modulazione della lunghezza di canale che non è nullo. Questa osservazione è confermata dal grafico della Figura 5-17, che mostra il tempo di propagazione dell'invertitore in funzione della tensione di alimentazione. Non deve sorprendere il fatto che questa curva ha la stessa forma di quella riportata nella Figura 3-28, che rappresenta la resistenza equivalente del transistor MOS in funzione di V_{DD}.

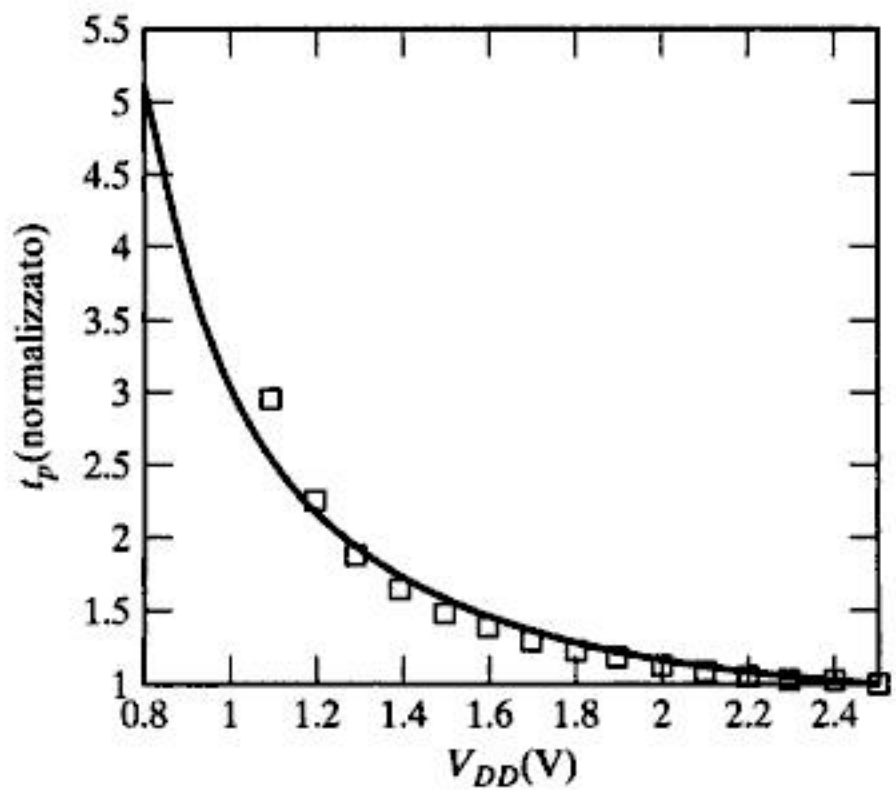


Figura 5-17 Tempo di propagazione di un invertitore CMOS in funzione della tensione di alimentazione (normalizzato rispetto al ritardo ottenuto per V_{DD} = 2.5 V). I simboli rappresentano i valori ricavati dall'Eq. (5.21). Osserviamo che quell'equazione è valida solo se entrambi i dispositivi lavorano in regione di saturazione di velocità. Questo spiega quindi la deviazione dalla curva simulata per bassi valori di tensione di alimentazione.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Rapporto tra NMOS e PMOS

Fino ad ora abbiamo sempre dimensionato il PMOS in modo tale che la sua resistenza fosse circa la stessa del transistor NMOS del pull-down. Ciò richiede, tipicamente, un rapporto tra le larghezze dei transistor PMOS e NMOS compreso tra 2 e 3. Questo approccio consente di ottenere un invertitore con caratteristica VTC abbastanza simmetrica e tempi di propagazione alto-basso e basso-alto simili. Ciò non significa comunque che questo rapporto fornisca il minimo valore del tempo di propagazione totale. Se i margini di rumore e la simmetria della VTC non sono di primaria importanza è effettivamente possibile ridurre la larghezza del PMOS in modo tale rendere l'invertitore più veloce.

Il ragionamento su cui si fonda questa affermazione è che un PMOS largo, se da un lato migliora il tempo di propagazione t_{pLH} dell'invertitore aumentando la corrente di carica, dall'altro penalizza il tempo di propagazione opposto t_{pHL} aumentando la capacità parassita. Dal momento che vi sono due esigenze con effetti contrastanti, deve esistere un rapporto ideale tra le dimensioni di PMOS e NMOS che ottimizza il tempo di propagazione globale.

Questo valore ottimo del rapporto può essere ricavato in modo abbastanza semplice. Consideriamo per esempio due invertitori CMOS identici in cascata. In prima approssimazione, la capacità di carico complessiva del primo invertitore è data da:

$$C_L = (C_{dp1} + C_{dn1}) + (C_{gp2} + C_{gn2}) + C_W \tag{5.23}$$

dove C_{dp1} e C_{dn1} sono le capacità equivalenti dei terminali di drain dei transistor PMOS e NMOS del primo invertitore e C_{gp2} e C_{gn2} sono le capacità dei gate dei transistor del secondo invertitore. C_W rappresenta invece la capacità delle piste di interconnessione.

Supponiamo che il dispositivo PMOS sia realizzato β volte più grande di quello NMOS ($\beta = (W/L)_p/(W/L)_n$). Quindi, tutte le capacità del PMOS scaleranno in modo approssimativamente lineare rispetto a quelle del transistor NMOS: $C_{dp1} \approx \beta\, C_{dn1}$ e $C_{gp2} \approx \beta\, C_{gn2}$. L'Eq. (5.23) può essere riscritta nel modo seguente:

$$C_L = (1 + \beta)(C_{dn1} + C_{gn2}) + C_W \tag{5.24}$$

Sostituendo questa espressione nell'Eq. (5.20) e ricordando che la resistenza equivalente di un transistor è inversamente proporzionale alla sua larghezza, arriviamo alla seguente espressione del tempo di propagazione totale:

$$\begin{aligned} t_p &= \frac{0{,}69}{2}((1+\beta)(C_{dn1} + C_{gn2}) + C_W)\left(R_{eqn} + \frac{R_{eqp}}{\beta}\right) \\ &= 0{,}345((1+\beta)(C_{dn1} + C_{gn2}) + C_W)R_{eqn}\left(1 + \frac{r}{\beta}\right) \end{aligned} \tag{5.25}$$

Qui, r ($= R_{eqp}/R_{eqn}$) rappresenta il rapporto tra le resistenze equivalenti di due transistor PMOS e NMOS con la stessa dimensione. Il valore ottimo di β può essere calcolato derivando il tempo di propagazione rispetto a β e uguagliando a 0 la derivata:

$$\frac{\partial t_p}{\partial \beta} = 0$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Dimensionamento e velocità dell'invertitore

Nell'analisi che segue considereremo un invertitore CMOS simmetrico, cioè un invertitore nel quale PMOS e NMOS sono dimensionati in modo tale da ottenere tempi di propagazione identici. Come già osservato la capacità di carico dell'invertitore può essere scomposta in due termini: $C_L = C_{int} + C_{ext}$. C_{int} rappresenta la *capacità intrinseca di uscita* dell'invertitore associata alle capacità delle diffusioni dei drain di NMOS e PMOS e alle due capacità di sovrapposizione gate–drain (riferite a massa per l'effetto Miller). C_{ext} è il *carico esterno*, attribuibile alle altre porte connesse in uscita (fan-out) e alla capacità parassita delle piste di interconnessione. Assumendo che la resistenza di uscita della porta logica sia R_{eq} possiamo esprimere il tempo di propagazione come

$$\begin{aligned} t_p &= 0{,}69 R_{eq}(C_{int} + C_{ext}) \\ &= 0{,}69 R_{eq} C_{int}(1 + C_{ext}/C_{int}) = t_{p0}(1 + C_{ext}/C_{int}) \end{aligned} \tag{5.27}$$

dove $t_{p0} = 0.69\ R_{eq}C_{int}$ rappresenta il tempo di ritardo di un invertitore caricato dalle sole capacità interne ($C_{ext} = 0$), ed è chiamato *tempo di ritardo (propagazione) intrinseco*.

La prossima domanda è: quale influenza ha il dimensionamento dei transistor sul tempo di propagazione dell'invertitore? Per rispondere dobbiamo definire una relazione tra i vari parametri nell'Eq. (5.27) e un fattore di scala S, che metta in relazione le dimensioni dei transistor con una data porta di riferimento, generalmente un invertitore di dimensione minima che ha capacità intrinseca di uscita C_{iref} e resistenza di uscita R_{ref}. La capacità intrinseca di uscita C_{int} consiste delle capacità delle diffusioni e la capacità di sovrapposizione gate-drain, entrambe le quali sono proporzionali alla larghezza dei transistor. Quindi, $C_{int} = SC_{iref}$. La resistenza di uscita della porta è legata alla resistenza dell'invertitore di riferimento dalla relazione $R_{eq} = R_{ref}/S$. Possiamo allora riscrivere l'Eq. (5.27) come

$$\begin{aligned} t_p &= 0{,}69(R_{ref}/S)(SC_{iref})(1 + C_{ext}/(SC_{iref})) \\ &= 0{,}69 R_{ref} C_{iref}\left(1 + \frac{C_{ext}}{SC_{iref}}\right) = t_{p0}\left(1 + \frac{C_{ext}}{SC_{iref}}\right) \end{aligned} \tag{5.28}$$

Da questa espressione possiamo trarre alcune conclusioni importanti:

- Il tempo di ritardo intrinseco t_{p0} è indipendente dalle dimensioni dell'invertitore ed è determinato solo dai parametri tecnologici e dal layout dell'invertitore stesso. Se in uscita non vi è nessun carico, l'aumento delle dimensioni dei transistor è completamente bilanciato dall'aumento corrispondente delle capacità parassite.
- Aumentando S all'infinito si ottengono le massime prestazioni, poiché si elimina il contributo delle capacità esterne e si riduce il tempo di ritardo a quello intrinseco. Se il fattore di scala S è abbastanza grande rispetto a (C_{ext}/C_{int}) un ulteriore aumento non migliora le prestazioni dell'invertitore, ma comporta solo un aumento del consumo di area.

Materiale protetto da copyright

Esempio 5.7 Dimensionamento dei dispositivi e velocità

Studiamo il miglioramento che possiamo ottenere in termini di velocità, dimensionando opportunamente i dispositivi dell'Esempio 5.5. Dai valori riportati nella Tabella 5-2 calcoliamo $C_{int}/C_{ext} \approx 1.05$ ($C_{int} = 3.0$ fF, $C_{ext} = 3.15$ fF). Ciò si traduce in un tempo di propagazione 2.05 volte maggiore rispetto al ritardo intrinseco. Usando un fattore di scala $S = 10$, possiamo ridurre il tempo di propagazione ad un valore maggiore solo del 10% rispetto al minimo teorico. Un ulteriore aumento di S produce solo un modesto miglioramento.

Tutto ciò è confermato dai risultati della simulazione mostrati nella Figura 5-19, che mostra il tempo di propagazione in funzione di S e prevede un miglioramento massimo delle prestazioni di un fattore 1.9 ($t_{p0} = 19.3$ ps). Nel grafico osserviamo che la maggior parte del beneficio è ottenuta con $S = 5$, mentre aumentare S a valori maggiori di 10 porta un beneficio modesto.

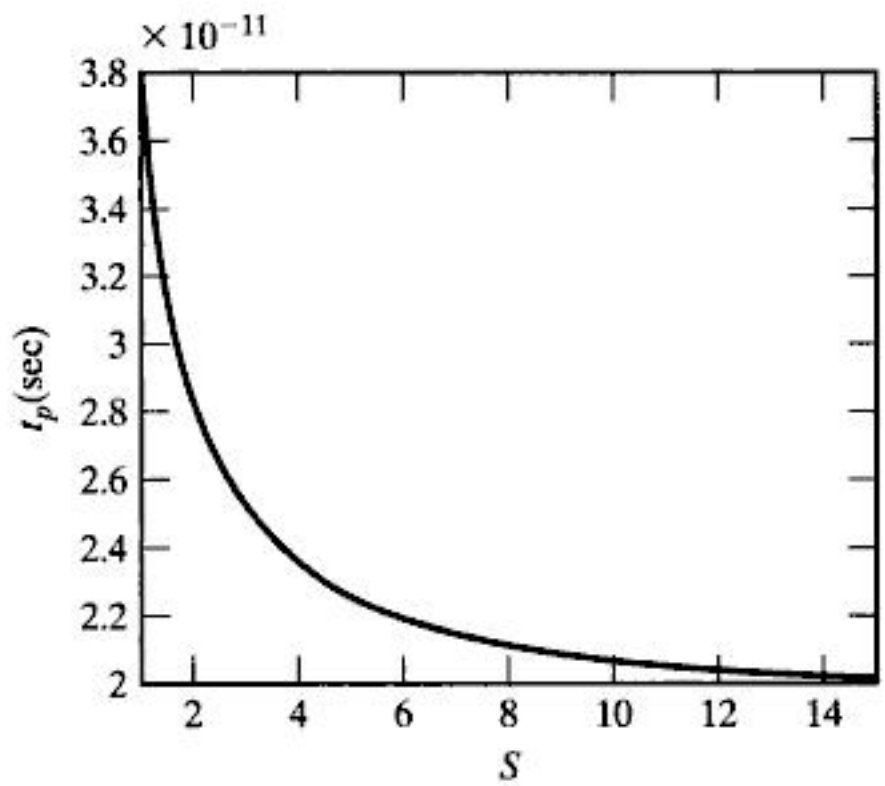


Figura 5-19 Prestazioni dell'invertitore di Figura 5-15 ottenute aumentando dello stesso fattore S le dimensioni dei transistor NMOS e PMOS per un valore fissato del fan-out.

Dimensionamento di una catena di invertitori

Se da un lato aumentare le dimensioni di un invertitore riduce il tempo di ritardo, dall'altro aumenta anche la capacità di ingresso. Dimensionare una porta logica senza curarsi delle ripercussioni sulla parte di circuito che la precede è un puro esercizio accademico. Un problema molto più interessante è determinare il dimensionamento ottimale di una porta logica quando è **inserita in un circuito reale**. Una semplice catena di invertitori costituisce un primo esempio interessante e rappresenta un modo per implementare un *buffer* CMOS. Innanzi tutto, per esprimere più comodamente l'effetto di carico introdotto da ciascun invertitore sul precedente, definiamo una relazione tra le capacità di ingresso C_g e la capacità intrin-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

l'Eq. (5.36) rispetto al numero di stadi N e uguagliando la derivata a 0. Procedendo in questo modo, otteniamo:

$$\gamma + \sqrt[N]{F} - \frac{\sqrt[N]{F}\ln F}{N} = 0$$

o equivalentemente:

$$f = e^{(1+\gamma/f)} \tag{5.37}$$

L'equazione (5.35) ha soluzione in forma chiusa solo se $\gamma = 0$, cioè quando si ignora l'effetto della capacità intrinseca di uscita e la capacità totale di carico coincide con quella associata al fan-out. In queste condizioni troviamo che il numero ottimale di stadi è $N = \ln(F)$, e il fan-out effettivo di ciascuno stadio è $f = e = 2.71828$. Questo tipo di approccio prevede il dimensionamento esponenzialmente crescente degli invertitori [Mead80]. Quando invece si tiene conto della capacità intrinseca delle porte logiche, l'Eq. può essere risolta solo numericamente e il risultato è mostrato in Figura 5-21a. Per un valore (tipico) $\gamma \approx 1$, il valore ottimale di f è circa 3.6. La Figura 5-21b riporta il tempo di propagazione attraverso la catena (normalizzato al tempo di propagazione ottimale) in funzione del fan-out complessivo per $\gamma = 1$. Scegliere un valore di fan-out leggermente superiore al valore ottimo non comporta un significativo peggioramento delle prestazioni della catena, ma riduce il numero di stadi richiesti e l'area necessaria per l'implementazione. È pratica comune **scegliere un fan-out ottimo di 4**. L'impiego di troppi stadi ($f < f_{opt}$), d'altra parte, ha ripercussioni negative sul ritardo e dovrebbe essere evitato.

Il procedimento appena illustrato può essere esteso a reti più complesse di invertitori con diverse configurazioni di fan-out. Un esempio di ciò è mostrato in Figura 5-22. Dobbiamo semplicemente modificare l'espressione della capacità esterna C_{ext} per includere il fan-out addizionale.

Esempio 5.8 Uso di un buffer CMOS pilotare capacità elevate

La Tabella 5-3 elenca i valori del rapporto $t_{p,opt}/t_{p0}$ per tre casi diversi: per un singolo invertitore, per due invertitori in cascata e per un buffer costituito da una catena ottimizzata di invertitori. Nella tabella sono riportati i risultati relativi a diversi valori del

Tabella 5-3 t_{opt}/t_{p0} in funzione di F per diverse configurazioni circuitali.

F	singolo invertitore	due invertitori	buffer CMOS
10	*11*	*8.3*	*8.3*
100	*101*	*22*	*16.5*
1000	*1001*	*65*	*24.8*
10 000	*10 001*	*202*	*33.1*

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

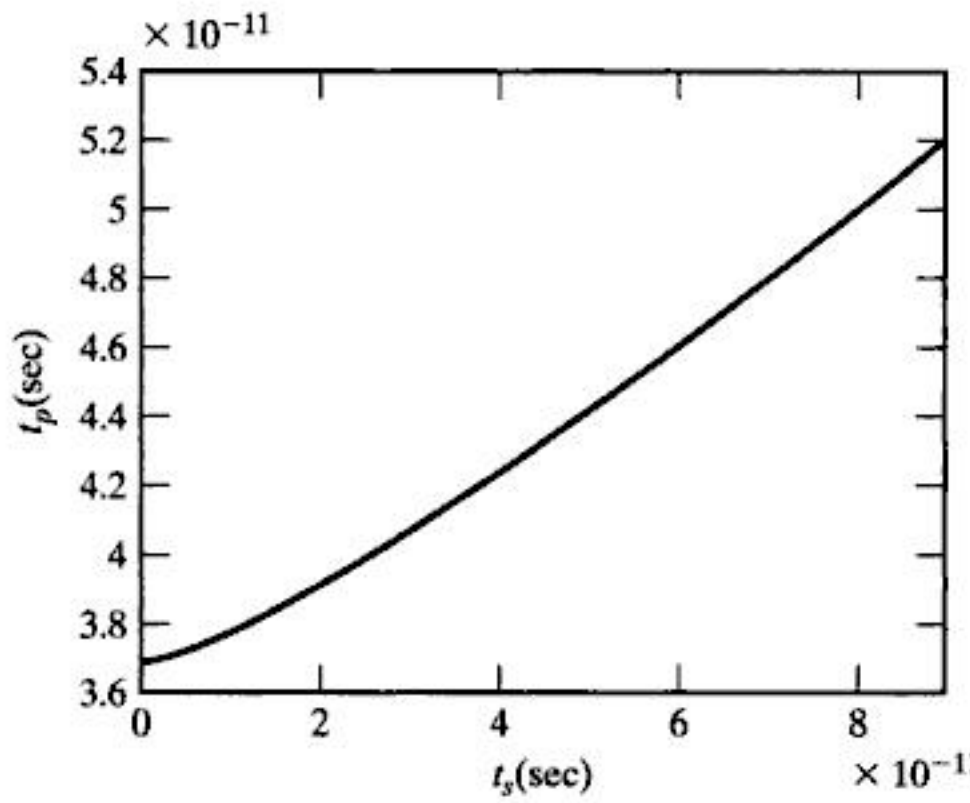


Figura 5-23 t_p in funzione del tempo di salita (discesa) del segnale di ingresso per un invertitore di dimensione minima che pilota un secondo invertitore identico.

effetti dei tempi di salita e discesa non nulli dell'ingresso direttamente alla loro origine, ossia la corrente limitata che può fornire l'uscita dello stadio precedente. Se quest'ultima fosse infinita (ovvero la porta avesse una resistenza di uscita nulla) la sua uscita commuterebbe istantaneamente, e lo stadio successivo vedrebbe un ingresso ideale, senza che il tempo di propagazione ne risultasse modificato. La potenzialità di questo tipo di approccio è che non considera la porta logica isolata e fine a se stessa, ma la inserisce in un contesto reale, analizzando le sue prestazioni in funzione sia del fan-out, sia degli stadi che la precedono. Ciò porta a un'espressione modificata del tempo di propagazione dell'invertitore i-esimo della catena [Hedenstierna87]:

$$t_p^i = t_{step}^i + \eta t_{step}^{i-1} \tag{5.38}$$

L'Eq. (5.38) afferma che il tempo di propagazione dell'invertitore i è uguale alla somma del suo tempo di propagazione ideale (t_{step}^i), cioè con pendenza infinita dell'ingresso, e di una frazione η del tempo di propagazione ideale dello stadio precedente ($i - 1$). η è una costante empirica, che vale circa 0.25. Questa espressione ha il vantaggio di essere molto semplice, pur considerando tutte le relazioni necessarie per l'analisi di circuiti complessi.

Esempio 5.9 Ritardo di un invertitore inserito in una catena

Prendiamo ad esempio il circuito della Figura 5-22. Mediante l'Eq. (5.31) e l'Eq. (5.38), possiamo calcolare l'espressione del tempo di propagazione dell'invertitore del secondo stadio della catena (disegnato nel riquadro grigio):

$$t_{p,2} = t_{p0}\left(1 + \frac{4C_{g,3}}{\gamma C_{g,2}}\right) + \eta t_{p0}\left(1 + \frac{4C_{g,2}}{\gamma C_{g,1}}\right)$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

$$
\begin{aligned}
t_p &= 0{,}69R_{dr}C_{int} + (0{,}69R_{dr} + 0{,}38R_w)C_w + 0{,}69(R_{dr} + R_w)C_{fan} \\
&= 0{,}69R_{dr}(C_{int} + C_{fan}) + 0{,}69(R_{dr}c_w + r_wC_{fan})L + 0{,}38r_wc_wL^2
\end{aligned}
\tag{5.39}
$$

Il fattore 0.38 tiene conto del fatto che che la pista è caratterizzata da una resistenza e una capacità distribuite. C_w e R_w rappresentano rispettivamente la capacità e la resistenza totale della pista. L'espressione del tempo di ritardo include sia un termine che dipende linearmente da L sia uno quadratico con L. È proprio a causa di quest'ultimo termine che il tempo di ritardo associato alle piste di interconnessioni sta diventando il fattore dominante nel budget complessivo, specialmente nel caso delle lunghe interconnessioni tra i blocchi.

Esempio 5.10 Tempo di ritardo dell'invertitore in presenza di lunghe interconnessioni

Consideriamo il circuito della Figura 5-24 e assumiamo validi gli stessi parametri utilizzati nell'Esempio 5.5: $C_{int} = 3$ fF, $C_{fan} = 3$ fF e $R_{dr} = 0.5(13/1.5 + 31/4.5) = 7.8$ kΩ. La pista è implementata in metal1 ed è larga 0.4 μm (la minima consentita). Da ciò ricaviamo i seguenti parametri dei parassiti della linea: $c_w = 92$ aF/μm e $r_w = 0.19$ Ω/μm (Esempio 4.4). Dall'Eq. (5.39) possiamo calcolare la lunghezza critica della pista per la quale il tempo di ritardo dovuto all'interconnessione diviene uguale al ritardo intrinseco causato dai soli parametri parassiti dei dispositivi. Risolvendo la seguente equazione di secondo grado otteniamo un'unica soluzione significativa:

$$
6{,}6 \times 10^{-18}L^2 + 0{,}5 \times 10^{-12}L = 32{,}29 \times 10^{-12}
$$

$$
\text{o, } L = 65\ \mu\text{m}
$$

Si può osservare che il ritardo aggiunto dalla linea di collegamento è attribuibile quasi esclusivamente al fattore lineare dell'equazione, in particolare, alla capacità parassita della pista. Il fattore quadratico (relativo al ritardo intrinseco della linea) comincia a dominare solo per piste molto più lunghe (> 7 cm). Ciò trae origine dalla resistenza propria dei transistor dell'invertitore, che hanno dimensione minima (o quasi). Una situazione molto diversa si avrebbe se l'invertitore fosse realizzato con dispositivi molto più larghi. Provare a svolgere lo stesso problema ma con transistor 100 volte più larghi.

5.5 Potenza, energia e prodotto energia-ritardo

Finora abbiamo visto che l'invertitore CMOS con la sua VTC quasi ideale (forma simmetrica, escursione logica piena tra massa e alimentazione ed elevati margini di rumore) offre una robustezza di elevata superiorità, che semplifica il flusso di progetto e apre le porte alla progettazione automatica. Un'altra qualità attraente dell'invertitore CMOS statico è la quasi totale assenza di consumo di potenza in condizioni stazionarie. È proprio questa combinazione tra robustezza e bassissimo consumo statico che hanno reso la tecnologia CMOS statica il cavallo di battaglia per la gran parte dei moderni circuiti digitali. Il consumo di potenza dei circuiti CMOS statici, invece, è dominato dal contributo dinamico proveniente dalla carica e scarica delle capacità del circuito.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Esempio 5.12 Fattore di attività

Consideriamo le forma d'onda in Figura 5-27, dove la forma d'onda superiore rappresenta il clock del circuito e quella inferiore mostra l'uscita del circuito. Solo 2 su 8 commutazioni del clock sono associate a una corrispondente transizione del segnale che dia luogo a consumo di potenza. Ciò equivale a una fattore di attività di 0.25 (25%).

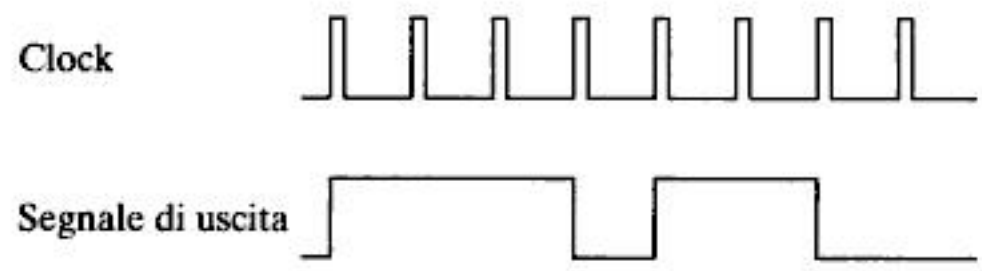


Figura 5-27 Forme d'onda del clock e del segnale di uscita.

Tecniche di progettazione per sistemi a basso consumo

Con la crescente complessità dei circuiti integrati digitali, si prevede che il consumo di potenza potrà solo aumentare nelle future tecnologie. Questa è una delle ragioni per cui soluzioni circuitali a bassa tensione di alimentazione stanno diventando sempre più attraenti. **Ridurre V_{DD} ha un effetto quadratico sul consumo dinamico**. Ridurre V_{DD} da 2.5 V a 1.25 V, per esempio, farebbe decrescere il consumo di potenza (relativo al precedente esempio) il consumo di potenza da 5 W a 1.25 W. Questo calcolo assume implicitamente che il circuito possa lavorare alla stessa frequenza di clock. La Figura 5-17 dimostra che questa ipotesi non è così illusoria se la tensione di alimentazione è sensibilmente maggiore della tensione di soglia, ma le prestazioni sono fortemente penalizzate non appena V_{DD} si avvicina troppo a 2 V_T.

Qualora il limite inferiore alla tensione di alimentazione sia fissato da vincoli esterni (come spesso accade nella realtà) o quando la diminuzione della tensione di alimentazione si traduce in un degrado inaccettabile per le prestazioni, il solo modo di diminuire il consumo è quello di ridurre la *capacità effettiva* di carico. Questo problema può essere affrontato agendo su entrambi i fattori che la determinano: la capacità fisicamente presente ai nodi e il fattore di attività.

La *riduzione del fattore di attività* è possibile solo a livello di astrazione logica e architetturale del sistema e sarà discusso in maggior dettaglio nel Capitolo 11. *Ridurre il valore fisico della capacità* è un obiettivo ugualmente importante e aiuta molto spesso a migliorare anche le prestazioni del circuito. Poiché la maggior parte delle capacità in un circuito logico derivano dai contributi parassiti dei transistor (gate e diffusioni), in fase di progettazione è ragionevole mantenere questi contributi al minimo valore possibile. Ciò significa realizzare i dispositivi con *dimensioni minime* ovunque sia possibile. Senza alcun dubbio ciò ha ripercussioni negative sulle prestazioni del circuito, ma tali effetti possono essere controbilanciati utilizzando tecniche e architetture logiche che permettano di accelerare il funzionamento del circuito. I soli casi in cui le dimensioni dei transistor devono essere necessariamente aumentate si hanno quando le capacità di carico sono dominate dai contributi esterni (fan-out o capacità delle piste). Questo tipo di approccio è in contraddizione con quello generalmente adottato nella progettazione delle librerie di celle standard, dove i transistor sono disegnati con dimensioni abbastanza grandi per garantire determinati requisiti in termini di prestazioni con carichi esterni anche abbastanza elevati.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

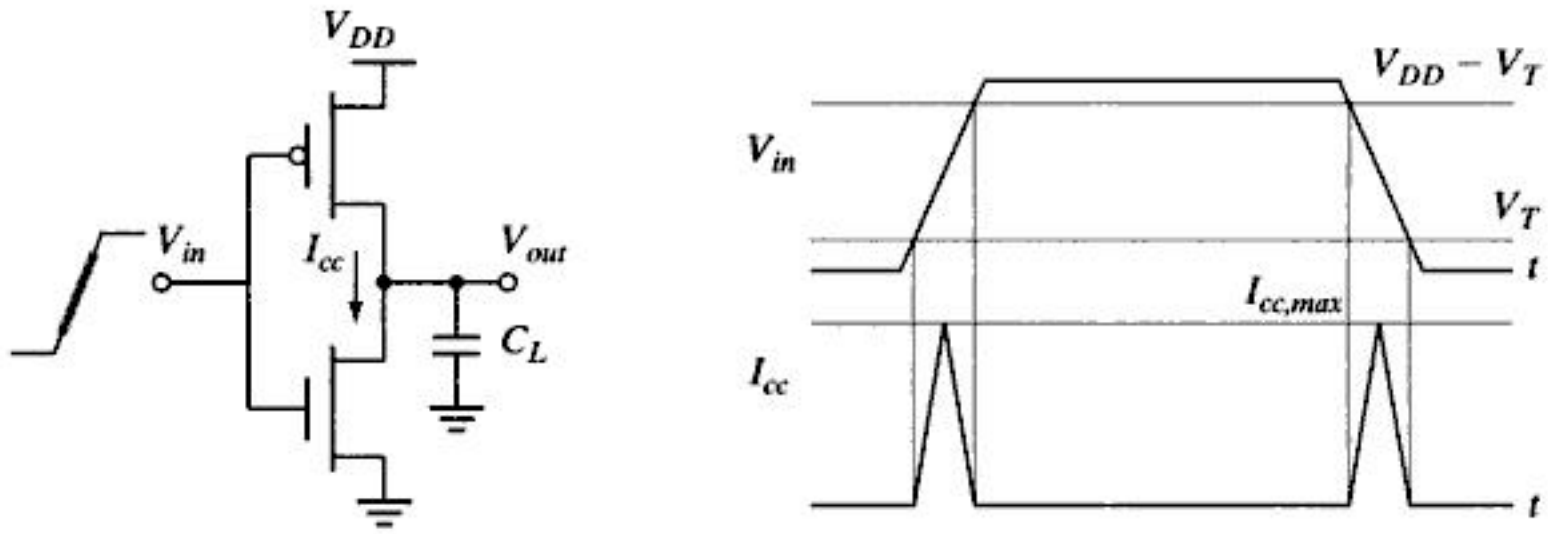


Figura 5-30 Corrente di cammino diretto (o di cortocircuito) durante la transizione.

t_{cc} rappresenta l'intervallo di tempo durante il quale entrambi i transistor sono simultaneamente accesi (tempo di cortocircuito) e I_{cc} l'intensità massima della corrente che attraversa i due MOS (corrente di cortocircuito o di picco). Possiamo quindi calcolare il consumo medio di potenza:

$$P_{cc} = t_{cc} V_{DD} I_{cc,max} f = C_{sc} V_{DD}^2 f \tag{5.50}$$

Il consumo di potenza per cammino diretto è proporzionale al fattore di attività del circuito, analogamente a quello dinamico. Nell'ipotesi che il segnale di ingresso abbia una pendenza costante durante la commutazione, t_{cc} può essere ragionevolmente approssimato con

$$t_{cc} = \frac{V_{DD} - 2V_T}{V_{DD}} t_s \approx \frac{V_{DD} - 2V_T}{V_{DD}} \times \frac{t_{r(f)}}{0{,}8} \tag{5.51}$$

dove t_s rappresenta il tempo necessario ad eseguire l'escursione completa (0–100%).

$I_{cc,max}$ è determinato dalla corrente di saturazione dei dispositivi ed è quindi direttamente proporzionale alla larghezza dei transistor. La corrente di picco è anche **funzione del rapporto tra le pendenze dei segnali di ingresso e di uscita**, come è possibile dedurre dalla semplice analisi che segue: consideriamo un invertitore CMOS statico il cui ingresso compie una commutazione $0 \rightarrow 1$. Supponiamo inizialmente che la capacità di carico in uscita sia molto grande, cosicché il tempo di discesa dell'uscita sia significativamente più grande del tempo di salita dell'ingresso (Figura 5-31a). In queste condizioni l'ingresso compirà la transizione completa da 0 V a V_{DD}, prima che la tensione di uscita cambi in modo apprezzabile. Poiché la tensione source-drain del transistor PMOS è circa nulla durante la commutazione dell'ingresso, il dispositivo si spegne velocemente senza essere attraversato da una corrente significativa. In questo caso la corrente di cortocircuito è praticamente nulla. Vediamo ora il caso opposto, ossia ipotizziamo che la capacità di uscita sia molto piccola e il tempo di discesa dell'uscita sia paragonabile o inferiore al tempo di salita dell'ingresso (Figura 5-31b). Per la maggior parte del transitorio della commutazione la tensione source-drain del PMOS è uguale a V_{DD}, dando luogo alla massima corrente di cortocircuito (uguale alla corrente di saturazione del PMOS). Quest'ultima rappresenta chiaramente la condizione di caso peggiore. Questo ragionamento è confermato dal grafico della Figura 5-32, che mostra la simula-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

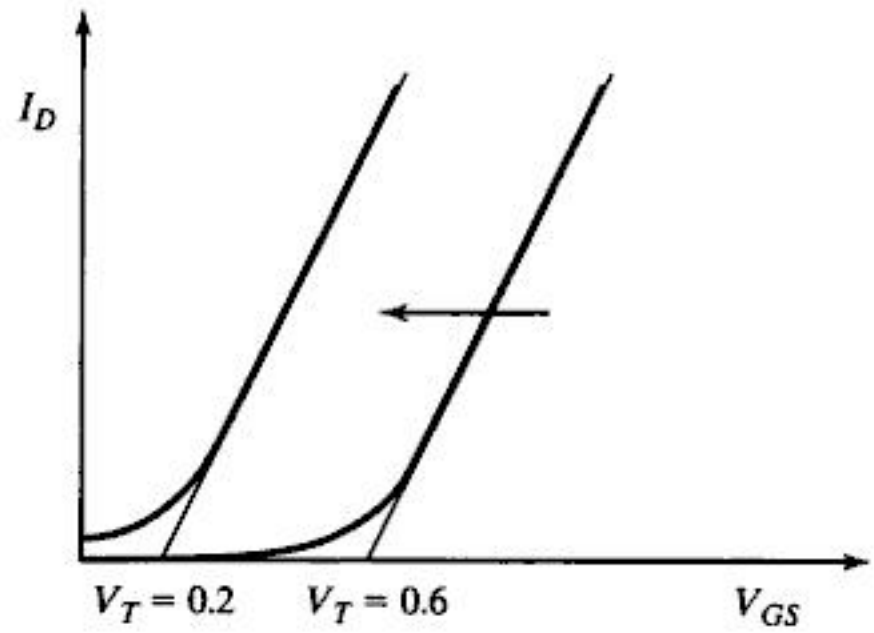


Figura 5-35 Diminuendo la tensione di soglia aumenta la corrente di sottosoglia a $V_{GS} = 0$.

= 0 V e maggiore è il consumo statico. Per contrastare questo effetto, la tensione di soglia è generalmente mantenuta abbastanza alta. I processi CMOS standard sono attualmente caratterizzati da tensioni di soglia di almeno 0.5–0.6 V e a volte anche maggiori (~ 0.75 V).

Questa tendenza è in conflitto con la riduzione della tensione di alimentazione che tipicamente è ridotta in accordo con l'evoluzione tecnologica, come appare chiaro dalla Figura 3-41. Abbiamo visto in precedenza (Figura 5-17) che ridurre la tensione di alimentazione, mantenendo invariata la tensione di soglia, si traduce in una sostanziale diminuzione delle prestazioni, specialmente quando V_{DD} si avvicina a 2 V_T. Un rimedio a questo problema consiste nel ridurre anche la tensione di soglia: ciò sposta la curva della Figura 5-17 verso sinistra, recuperando così la perdita in prestazioni che si aveva con la riduzione di V_{DD}. Sfortunatamente, la riduzione della tensione di soglia ha un limite inferiore dettato dal massimo consumo statico consentito, come dimostrato nella Figura 5-35. La scelta della tensione di soglia rappresenta quindi un importante compromesso tra prestazioni e consumo. Tuttavia, la riduzione continua della tensione di alimentazione prevista per le prossime generazioni tecnologiche CMOS, sta spingendo la tensione di soglia a valori sempre più bassi, rendendo la corrente di sottosoglia una fonte non più trascurabile di consumo di potenza. Alla luce di ciò, si comprende perché nel futuro sarà sempre più attraente il passaggio verso processi tecnologici in grado di realizzare dispositivi caratterizzati da una maggior pendenza della corrente in regione di sottosoglia. Un esempio ne è la tecnologia SOI (Silicon-on-Insulator) in cui la corrente di sottosoglia dei transistor MOS ha una pendenza molto vicina al valore ideale di 60 mV/decade.

Esempio 5.14 Tensione di soglia, prestazioni e consumo statico di potenza

Consideriamo un transistor NMOS di dimensione minima in una tecnologia CMOS da 0.25 μm. Nel Capitolo 3 abbiamo calcolato che la pendenza della corrente di sottosoglia è $S = 90$ mV/decade. La corrente del transistor in sottosoglia con $V_{GS} = 0$ e V_T *approssimativamente di* 0.5 V è uguale a 10^{-11} A (Figura 3-22). Riducendo di 200 mV la tensione di soglia (fino a 0.3 V) la corrente di sottosoglia aumenta di un fattore 170. Supponiamo di avere un circuito con un milione di porte logiche e ali-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

prestazioni, che richiedono una tensione di alimentazione maggiore a spese del consumo. Analogamente, un sistema a basso consumo deve lavorare a tensioni di alimentazione più basse, garantendo le necessarie prestazioni mediante l'uso di architetture particolari quali la pipeline o il parallellismo.

5.5.4 Analisi del consumo di potenza con SPICE

La definizione del consumo medio di potenza è già stata fornita nel Capitolo 1, e viene qui ripetuta per comodità:

$$P_{av} = \frac{1}{T}\int_0^T p(t)dt = \frac{V_{DD}}{T}\int_0^T i_{DD}(t)dt \tag{5.60}$$

dove T è l'intervallo di interesse e V_{DD} e i_{DD} sono rispettivamente la tensione di alimentazione e la corrente assorbita dall'alimentazione. Alcune implementazioni del simulatore SPICE incorporano delle funzioni per misurare il valore medio di un segnale nel circuito. Per esempio, nel simulatore HSPICE l'istruzione ".MEASURE TRAN I(VDD) AVG" esegue il calcolo dell'area sottesa dalla curva corrispondente a un particolare segnale (I(VDD) in questo caso) e divide il risultato per il periodo di interesse. Questo calcolo è identico alla definizione data nell'Eq. (5.60). Purtroppo altre implementazioni di SPICE non sono così evolute, ma ciò non è così drammatico come sembra: basta ricordare che SPICE non è nient'altro che un risolutore di equazioni differenziali. Possiamo implementare un piccolo circuito che funzioni da integratore, in modo tale che il segnale di uscita coincida con la potenza media.

Consideriamo, per esempio, il circuito disegnato nella Figura 5-37. La corrente fornita dall'alimentazione è misurata da un generatore di corrente controllato in corrente e integrata tramite il condensatore C; la resistenza R è usata solo per garantire la convergenza numerica e dovrebbe essere più alta possibile per ridurre la corrente di perdita. La scelta opportuna dei parametri degli elementi circuitali assicura che la tensione di uscita sia effettivamente la potenza media P_{av} del circuito di partenza. Nell'ipotesi che il valore iniziale della tensione di uscita sia zero, la funzione svolta dal circuito è sintetizzata nella seguente equazione differenziale:

$$C\frac{dP_{av}}{dt} = ki_{DD}$$

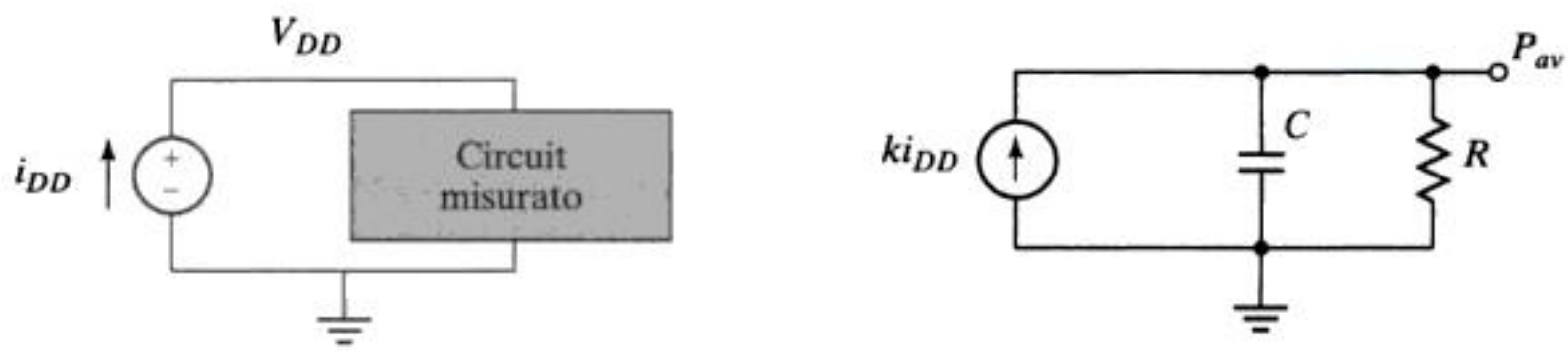


Figura 5-37 Circuito equivalente per la misura del consumo di potenza mediante SPICE.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

l'evoluzione sarà maggiormente in linea con lo scaling a campo costante, cioè con una densità di potenza costante. Anche in queste condizioni, tuttavia, il consumo di potenza per un singolo chip continuerà ad aumentare grazie alle dimensioni crescenti dell'area del chip.

Il modello evolutivo presentato ha comunque un sostanziale difetto. Le previsioni sulle prestazioni e sul consumo di potenza sono numeri che tengono conto solo dei parametri intrinseci dei dispositivi. Nel Capitolo 4 abbiamo visto che le interconnessioni evolvono secondo modelli sostanzialmente diversi e che gli elementi parassiti delle piste potrebbero presto dominare sulle prestazioni del circuito. Analogamente, caricare e scaricare le capacità delle piste potrebbe essere il fattore dominante sul consumo di energia. Per ottenere una previsione più incisiva e realistica, si dovrebbe costruire un modello di scaling che tenga conto dell'evoluzione dei dispositivi e delle interconnessioni simultaneamente. L'impatto delle capacità parassite delle piste e il loro andamento con l'evoluzione della tecnologia è riassunto nella Tabella 5-5. Abbiamo adottato il modello a resistenza fissa introdotto nel Capitolo 4 e abbiamo assunto che la resistenza di uscita delle porte logiche sia dominante sulla resistenza della pista (tipico delle piste medio-lunghe).

Il modello prevede che il tempo di ritardo (e il consumo) causato dalle interconnessioni sarà sempre più rilevante con il procedere dello scaling. In presenza di piste corte ($S = S_L$) l'aumento procede a una velocità limitata ε_c, ma la situazione peggiora con interconnessioni medio-lunghe ($S_L < S$). Queste considerazioni sono state confermate da un certo numero di studi, un esempio dei quali è mostrato dalla Figura 5-41. Il modo in cui evolverà il rapporto tra il tempo di ritardo (consumo) prodotto dalle piste e quello intrinseco delle porte logiche è ancora oggetto di dibattito, poiché dipende da vari parametri indipendenti, quali l'architettura del sistema, le metodologie progettuali, il dimensionamento dei transistor e i materiali usati per le interconnessioni. Certamente la visione apocalittica, in cui le prestazioni dei circuiti CMOS saturerebbero in un futuro non troppo lontano a causa delle interconnessioni, è sicuramente esagerata. Tuttavia appare chiara la necessità di guardare alle interconnessioni con sempre maggiore cura ed attenzione: ciò potrebbe presto cambiare le metodologie di progettazione e ottimizzazione dei circuiti nelle prossime generazioni (es. [Sylvester98]).

5.7 Riepilogo

In questo capitolo è stata effettuata l'analisi rigorosa e dettagliata dell'invertitore CMOS statico. Le caratteristiche salienti di questa porta logica possono essere così riassunte:

- L'invertitore CMOS statico è costituito da un transistor NMOS (pull-down) e un transistor PMOS (pull-up) che generalmente è più largo di quello NMOS poiché la sua conducibilità è intrinsecamente minore.
- La caratteristica di trasferimento ingresso-uscita è quasi ideale: l'escursione logica è massima tra 0 e V_{DD} e indipendente dalle dimensioni dei transistor; i margini di rumore di un invertitore simmetrico (dove PMOS e NMOS hanno la stessa resistenza equivalente) hanno valori vicini a $V_{DD}/2$. La caratteristica statica non è influenzata dal fan-out.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

elementare. Un ampia lista di riferimenti è stata presentata nel Capitolo 1. Di seguito riportiamo alcuni riferimenti di particolare interesse che sono stati citati in questo capitolo.

Riferimenti scientifici

[Dally98] W. Dally and J. Poulton, *Digital Systems Engineering*, Cambridge University Press, 1998.

[Fisher98] P. D. Fisher and R. Nesbitt, "The Test of Time: Clock-Cycle Estimation and Test Challenges for Future Microprocessors," *IEEE Circuits and Devices Magazine*, 14(2), pp. 37–44, 1998.

[Hedenstierna87] N. Hedenstierna and K. Jeppson, "CMOS Circuit Speed and Buffer Optimization," *IEEE Transactions on CAD*, vol. CAD-6, no. 2, pp. 270–281, March 1987.

[Kuroda95] T. Kuroda and T. Sakurai, "Overview of low-power ULSI circuit techniques," *IEICE Trans. on Electronics*, vol. E78-C, no. 4, pp. 334-344, April 1995.

[Liu93] D. Liu and C. Svensson, "Trading speed for low power by choice of supply and threshold voltages," *IEEE Journal of Solid-State Circuits*, vol. 28, no.1, pp. 10–17, Jan. 1993, p.10–17.

[Mead80] C. Mead and L. Conway, *Introduction to VLSI Systems*, Addison-Wesley, 1980.

[Sedra87] A. Sedra and K. Smith, *MicroElectronic Circuits*, Holt, Rinehart and Winston, 1987.

[Swanson72] R. Swanson and J. Meindl, "Ion-Implanted Complementary CMOS transistors in Low-Voltage Circuits," *IEEE Journal of Solid-State Circuits*, vol. SC-7, no. 2, pp.146–152, April 1972.

[Sylvester98] D. Sylvester and K. Keutzer, "Getting to the Bottom of Deep Submicron," *Proceedings ICCAD Conference*, pp. 203, San Jose, November 1998.

[Veendrick84] H. Veendrick, "Short-Circuit Dissipation of Static CMOS Circuitry and its Impact on the Design of Buffer Circuits," *IEEE Journal of Solid-State Circuits*, vol. SC-19, no. 4, pp. 468–473, 1984.

Esercizi e progetti

Consultare la pagina web **http://bwrc.eecs.berkeley.edu/IcBook** per una lista aggiornata di problemi ed esercizi. Rendendo disponibili gli esercizi in formato elettronico anziché stampati, siamo in grado di garantire il continuo aggiornamento e seguire la rapida evoluzione delle attuali tecnologie e dei metodi di progettazione dei circuiti integrati digitali.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

6.1 Introduzione

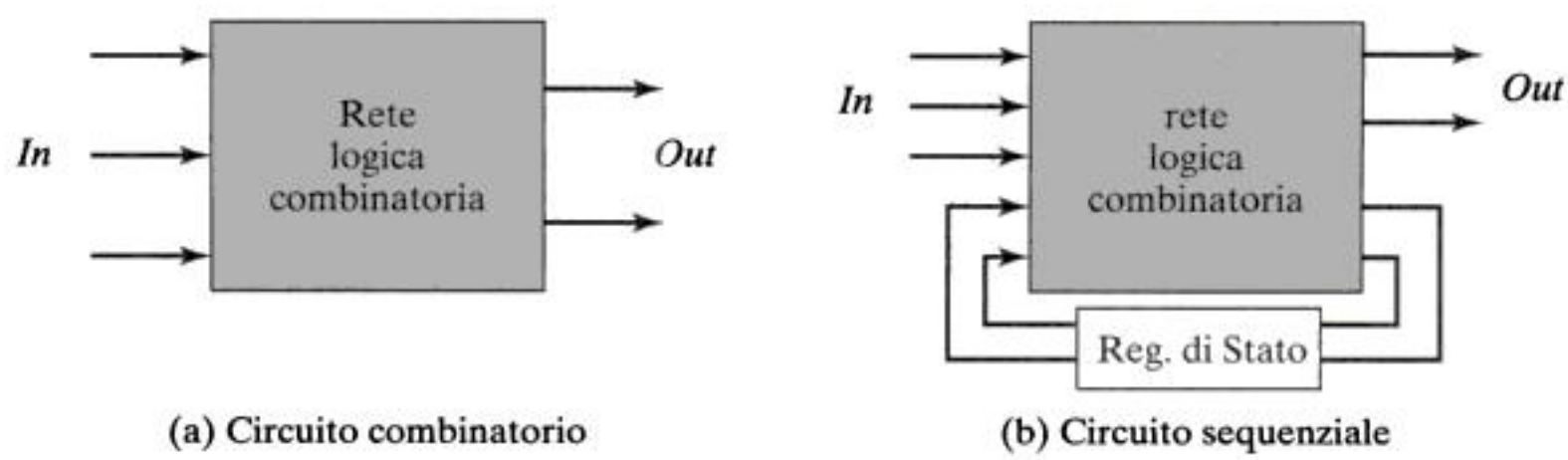
Il capitolo precedente è stato dedicato allo studio dell'invertitore CMOS statico. In questo capitolo riprenderemo tutte le considerazioni progettuali fatte in precedenza per affrontare la sintesi di porte logiche arbitrarie, come NOR, NAND e XOR. Argomento di questo capitolo saranno i circuiti logici *combinatori* o circuiti *senza memoria*, cioè tutti quei circuiti in cui ad ogni istante il valore logico dell'uscita è determinato da un'espressione booleana dei soli valori correnti degli ingressi (assumendo come trascurabili i tempi di propagazione attraverso le singole porte logiche) senza che vi sia nessuna retroazione del segnale di uscita.

Questa proprietà è ciò che li distingue da un'altra classe di circuiti, noti come circuiti *sequenziali* o circuiti *con memoria*, nei quali l'uscita è funzione non solo degli ingressi correnti, ma anche dei valori precedenti (vedi Figura 6-1). La funzione di *memoria* può essere realizzata connettendo una o più uscite a uno o più ingressi. Di conseguenza, il circuito "ricorda" gli eventi passati e ne conserva *memoria storica.* Un circuito sequenziale comprende una parte di logica combinatoria e un modulo che mantiene traccia dello stato precedente. Esempi di questi circuiti sono i registri, i contatori, gli oscillatori e le memorie. I circuiti sequenziali costituiscono l'argomento del prossimo capitolo.

Ci sono molti stili di implementazione delle funzioni logiche. Come per l'invertitore, le figure di merito comunemente usate per valutare le prestazioni di una porta sono l'area, la velocità, il consumo di potenza e l'energia richiesta per operazione. A seconda dell'applicazione specifica, avrà maggior rilievo l'una o l'altra di queste grandezze. Il parametro chiave per un processore ad alte prestazioni, per esempio, è la velocità di calcolo, mentre per un circuito alimentato attraverso batterie ha più importanza il consumo di energia. Recentemente il consumo di potenza sta diventando uno dei problemi più preoccupanti e sempre maggior enfasi viene usata per studiarne l'origine e identificare le metodologie progettuali per contenerlo. Oltre a queste metriche, la robustezza al rumore e l'affidabilità sono ugualmente importanti. Vedremo che alcune architetture circuitali possono migliorare apprezzabilmente le prestazioni, ma a discapito di una maggiore sensibilità ai disturbi.

6.2 Logica CMOS statica

La famiglia logica di gran lunga più usata è la *logica CMOS statica complementare*, che è la naturale estensione dell'invertitore CMOS statico per realizzare porte logiche più complesse e con un maggior numero di ingressi. Ricordiamo che i maggiori vantaggi dell'approccio sta-



(a) Circuito combinatorio (b) Circuito sequenziale

Figura 6-1 Classificazione di alto livello dei circuiti logici.

Materiale protetto da copyright

tico sono la robustezza (cioè la bassa sensibilità ai disturbi), le elevate prestazioni e il basso consumo con dissipazione nulla in condizioni stazionarie. Molte di queste proprietà si estendono anche alle porte con un fan-in maggiore, implementate con una topologia simile.

I circuiti CMOS complementari ricadono all'interno di una classe più ampia di architettura logica, detta *logica statica*, nella quale **ad ogni istante temporale, ciascun nodo di uscita è connesso o a V_{DD} o a *massa* attraverso un cammino a bassa resistenza**. Inoltre, in ciascun istante la tensione di uscita di ciascuna porta logica corrisponde sempre al valore booleano della funzione implementata dal circuito stesso (ignorando i transistori durante le commutazioni da uno statico all'altro). Questa caratteristica distingue l'architettura statica da quella *dinamica*, in cui il segnale logico di uscita consiste nella carica elettrica immagazzinata nella capacità parassita di un nodo del circuito in condizioni di alta impedenza. Questo secondo approccio permette di realizzare porte logiche più snelle e più veloci. Tuttavia il progetto di un circuito dinamico è molto più delicato a causa della maggior sensibilità ai disturbi, che può condurre al malfunzionamento della porta logica.

In questo paragrafo tratteremo la progettazione di vari tipi di circuiti statici, in particolare: la *logica statica complementare*, la *logica a rapporto*, (*pseudo-NMOS* e *DCVSL*) e la logica a *pass-transistor*. Prenderemo in considerazione anche le questioni e i problemi legati all'evoluzione delle tecnologie, alla riduzione della tensione di alimentazione e della tensione di soglia.

6.2.1 Logica CMOS statica complementare

Concetti fondamentali

Una porta CMOS statica complementare nasce dalla combinazione di due reti di transistor: la *rete di pull-up* (*pull-up network*, PUN) e la *rete di pull-down* (*pull-down network*, PDN). La Figura 6-2 mostra una generica porta logica a N ingressi. Tutti gli N ingressi sono distribuiti sia alla rete di *pull-up* sia a quella di *pull-down*. Il PUN ha lo scopo di connettere il nodo di uscita a V_{DD} quando la configurazione degli ingressi è tale che il valore logico della funzione booleana sia 1. Analogamente, il PDN fornisce una connessione tra l'uscita del circuito a *massa* quando la porta logica deve produrre 0 all'uscita. Le reti del *pull-up* e del *pull-down* sono costruite in modo mutuamente esclusivo nel senso che in condizioni stazionarie *una e solo una* delle due reti deve essere attiva. In questo modo, una volta che il transitorio della commutazione si è esaurito e gli ingressi sono stabili, esiste sempre un solo cammino conduttivo a bassa resistenza tra il nodo di uscita e l'alimentazione V_{DD} per il valore logico alto ("1"), oppure tra il nodo di uscita e la linea di *massa* per il valore logico basso ("0"). Ciò è equivalente all'affermare che il nodo di uscita è sempre un nodo a *bassa impedenza* in condizioni stazionarie.

Nel costruire la rete di transistor del PDN e del PUN, il progettista deve tenere in mente le osservazioni seguenti:

- Il transistor può essere pensato come un interruttore controllato dal segnale di gate. Un transistor (interruttore) NMOS è *attivo (chiuso)* quando il segnale di controllo è alto,

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Esempio 6.1 Porta NAND a due ingressi

La Figura 6-5 mostra una porta NAND a due ingressi ($F = \overline{A \cdot B}$). La rete del PDN consiste nella serie di due transistor che conduce solo quando A e B sono contemporaneamente alti. La rete di PUN è la rete duale e consiste nel parallelo dei due transistor PMOS. Ciò significa che F è 1 se $A = 0$ o $B = 0$, che è equivalente a $F = \overline{A \cdot B}$. La tabella di verità per la semplice porta NAND a due ingressi è riportata nella Tabella 6-1. Possiamo verificare che l'uscita F è sempre connessa o a V_{DD} o massa, ma mai ad entrambe allo stesso tempo.

Tabella 6-1 Tabella di verità di una NAND a 2 ingressi.

A	*B*	*F*
0	0	1
0	1	1
1	0	1
1	1	0

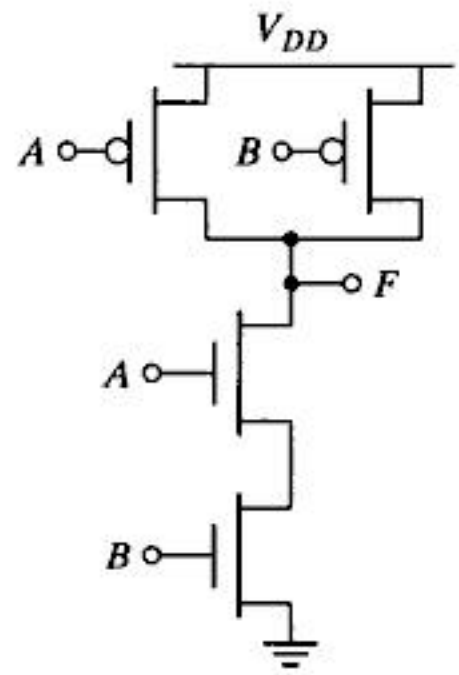


Figura 6-5 Porta NAND a due ingressi in logica CMOS statica complementare.

Esempio 6.2 Sintesi di una porta logica CMOS arbitraria

Utilizzando la logica CMOS statica complementare, realizziamo il circuito che svolge la funzione booleana $F = \overline{D + A \cdot (B + C)}$. Il primo passo per la sintesi della porta logica consiste nel derivare la rete di *pull-down*, come mostrato nella Figura 6-6a, ricordando il fatto che la serie di NMOS realizza la funzione AND e il parallelo di NMOS corrisponde alla funzione OR. Il passo successivo è quello di derivare il PUN applicando il principio di dualità ricorsivamente a tutte le sottoreti appena disegnate: la rete del PDN viene spezzata in quattro sottoreti più piccole (SR1, SR2, SR3 e SR4) che semplificano

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

tre casi possibili per caricare il nodo di uscita a V_{DD}. Se entrambi gli ingressi sono mantenuti bassi, i due dispositivi PMOS sono entrambi accesi e in questo caso il tempo di ritardo è $0.69 \times (R_p/2) \times C_L$, dato che abbiamo due resistenze in parallelo. Questo non è il caso peggiore per la commutazione basso-alto dell'uscita, che avviene anche quando uno solo dei transistor PMOS è acceso, dando luogo ad un ritardo di $0.69 \times R_p \times C_L$. Per quanto riguarda la rete di *pull-down*, l'uscita è scaricata a massa se e solo se entrambi i transistor sono accesi simultaneamente (A e B entrambi alti) e il tempo di propagazione è dato da $0.69 \times (2R_N) \times C_L$ in prima approssimazione. In altre parole, aggiungere transistor in serie ha come effetto un aumento del tempo di ritardo del circuito e i dispositivi devono essere realizzati più larghi per evitare di penalizzare troppo la risposta dinamica. Quando studiamo il dimensionamento dei transistor di una porta logica con due o più ingressi, dobbiamo sempre tenere presente la combinazione degli ingressi a cui corrisponde il caso peggiore per il tempo di propagazione.

Se desideriamo che la porta NAND appena considerata abbia il tempo di propagazione alto-basso (t_{pHL}) uguale a quello di un invertitore con dimensione minima, i transistor NMOS del PDN devono essere realizzati con larghezza doppia rispetto all'invertitore in modo tale che la resistenza equivalente del PDN della porta NAND sia la stessa dell'invertitore. Invece i dispositivi PMOS possono rimanere inalterati.[1]

Questa analisi al primo ordine ignora l'effetto dell'aumento delle capacità parassite dovute all'allargamento dei transistor. Nella pratica questa ipotesi non è quasi mai verificata, ma consente tuttavia un primo dimensionamento dei transistor, sebbene approssimativo.

Esempio 6.3 Dipendenza del tempo di propagazione dalla configurazione degli ingressi

Consideriamo la porta NAND della Figura 6-8a e assumiamo che i transistor NMOS e PMOS abbiano fattore di forma rispettivamente 0.5 μm/0.25 μm e 0.75 μm/0.25 μm. Questo dimensionamento dovrebbe fornire tempi di propagazione basso-alto e alto-basso approssimativamente uguali (essendo le resistenze effettive del *pull-down* e del *pull-up* nominalmente uguali).

La Figura 6-9 mostra il tempo di propagazione basso-alto simulato per diverse configurazioni dei segnali di ingresso. Naturalmente nel caso in cui entrambi gli ingressi commutano dal valore logico alto a quello basso ($A = B = 1 \rightarrow 0$) corrisponde il tempo di ritardo minore, in confronto ai casi in cui uno solo degli ingressi commuta mentre l'altro è tenuto costante a 1. Notiamo che il caso peggiore per il tempo di propagazione basso-alto dipende da quale dei due ingressi (A o B) è basso. La ragione di questa differenza risiede nella capacità del nodo interno della rete di *pull-down* (cioè, il source di M_2). Nel caso in cui $B = 1$ e A commuta da 1 a 0, il transistor PMOS del *pull-up* deve caricare solo la capacità del nodo di uscita (M_2 è spento). Al contrario, nel caso

[1] Nei processi CMOS submicrometrici, i transistor connessi in serie risentono meno della saturazione della velocità e non devono essere disegnati necessariamente così larghi. In una porta NAND a due ingressi potrebbe essere sufficiente realizzare i due dispositivi NMOS 1.5 volte più grandi (invece di 2 volte) rispetto al transistor NMOS di un invertitore.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

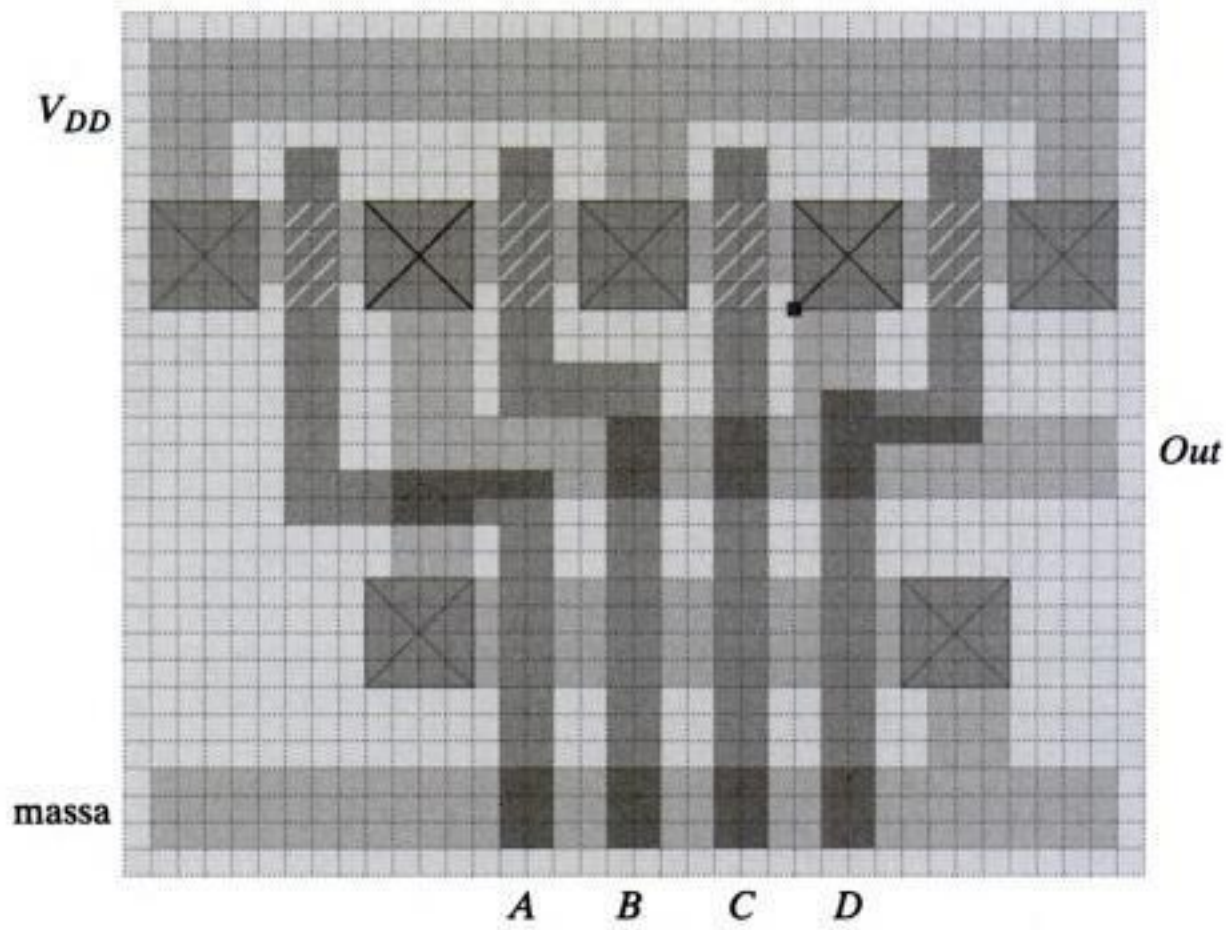


Figura 6-12 Layout di una porta NAND a quattro ingressi in logica CMOS complementare. Vedi anche l'inserto a colori numero 7.

Con una tecnica simile a quella usata nel Capitolo 5 per l'invertitore, possiamo calcolare le capacità dei nodi interni della porta NAND a partire dal suo layout. Osserviamo che nel *pull-up* i dispositivi PMOS condividono i terminali di drain a due a due, in modo da rendere minimo il contributo delle capacità parassite. Utilizzando le nostre regole standard di layout, calcoliamo facilmente l'area e il perimetro di ogni dispositivo. I risultati sono elencati nella Tabella 6-2.

Tabella 6-2 Area e perimetro dei transistor in una porta NAND a quattro ingressi.

Transistor	W (μm)	AS (μm²)	AD (μm²)	PS (μm)	PD (μm)
1	0.5	0.3125	0.0625	1.75	0.25
2	0.5	0.0625	0.0625	0.25	0.25
3	0.5	0.0625	0.0625	0.25	0.25
4	0.5	0.0625	0.3125	0.25	1.75
5	0.375	0.297	0.172	1.875	0.875
6	0.375	0.172	0.172	0.875	0.875
7	0.375	0.172	0.172	0.875	0.875
8	0.375	0.297	0.172	1.875	0.875

Materiale protetto da copyright

Tabella 6-3 Calcolo delle capacità parassite per una transizione alto-basso del nodo di uscita. La tabella mostra nell'ultima colonna il valore numerico in assenza di carico esterno. Qualsiasi capacità associata al *fan-out* esterno dovrebbe essere semplicemente aggiunta al termine C_L.

Capacità	Contributi (alto → basso)	Valore (fF) (alto → basso)
C_1	$C_{d1} + C_{s2} + 2\,C_{gd1} + 2\,C_{gs2}$	(0.57 * 0.0625 * 2 + 0.61 * 0.25 * 0.28) + (0.57 * 0.0625 * 2 + 0.61 * 0.25* 0.28) + 2 * (0.31 * 0.5) + 2 * (0.31 * 0.5) = 0.85 fF
C_2	$C_{d2} + C_{s3} + 2\,C_{gd2} + 2\,C_{gs3}$	(0.57 * 0.0625 * 2 + 0.61 * 0.25 * 0.28) + (0.57 * 0.0625 * 2 + 0.61 * 0.25* 0.28) + 2 * (0.31 * 0.5) + 2 * (0.31 * 0.5) = 0.85 fF
C_3	$C_{d3} + C_{s4} + 2\,C_{gd3} + 2\,C_{gs4}$	(0.57 * 0.0625 * 2+ 0.61 * 0.25 * 0.28) + (0.57 * 0.0625 * 2+ 0.61 * 0.25* 0.28) + 2 * (0.31 * 0.5) + 2 * (0.31 * 0.5) = 0.85 fF
C_L	$C_{d4} + 2\,C_{gd4} + C_{d5} + C_{d6} + C_{d7} + C_{d8} + 2\,C_{gd5} + 2\,C_{gd6} + 2\,C_{gd7} + 2\,C_{gd8} = C_{d4} + 2\,C_{gd4} + 4\,C_{d5} + 8\,C_{gd6}$	(0.57 * 0.3125 * 2 + 0.61 * 1.75 *0.28) + 2 * (0.31 * 0.5)+ 4 * (0.79 * 0.171875* 1.9+ 0.86 * 0.875 * 0.22)+ 4 * 2 * (0.27 * 0.375) = 3.47 fF

In questo esempio concentriamo la nostra attenzione sul tempo di ritardo del *pull-down* e le capacità sono calcolate per una transizione alto-basso dell'uscita. Sebbene l'uscita compia una commutazione da V_{DD} a 0, l'escursione dei nodi interni è limitata tra $V_{DD} - V_{Tn}$ a 0. Sarebbe dunque necessario linearizzare le capacità di queste giunzioni tenendo conto dell'escursione effettiva ma, per semplicità, useremo anche per queste lo stesso coefficiente K_{eff} della capacità del nodo di uscita.

Assumiamo che l'uscita della porta sia connessa a un singolo invertitore di dimensione minima e che l'effetto delle piste di interconnessione sia trascurabile. I diversi contributi sono riassunti nella Tabella 6-3. Per le giunzioni associate ai transistor abbiamo usato le coppie di coefficienti $K_{eq} = 0.57$ e $K_{eqsw} = 0.61$ (NMOS) e $K_{eq} = 0.79$ e $K_{eqsw} = 0.86$ (PMOS). Notiamo che la capacità gate-drain è stata moltiplicata per un fattore 2 per tutti i nodi interni così come per il nodo di uscita, per tenere conto dell'effetto Miller (trascurando il fatto che l'escursione dei nodi interni è leggermente minore a causa della caduta di tensione dovuta alla soglia dei transistor).

Usando l'Eq. (6.4), calcoliamo il tempo di propagazione:

$$t_{pHL} = 0{,}69\left(\frac{13\,\mathrm{K\Omega}}{2}\right)(0{,}85\ \mathrm{fF} + 2 \cdot 0{,}85\ \mathrm{fF} + 3 \cdot 0{,}85\ \mathrm{fF} + 4 \cdot 3{,}47\ \mathrm{fF}) = 85\ \mathrm{ps}$$

Il tempo di ritardo simulato per questa particolare commutazione è 86 ps. L'analisi manuale fornisce una stima molto accurata considerando tutte le semplificazioni e le linearizzazioni che abbiamo usato. Abbiamo, per esempio, assunto che le capacità

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

seca di uscita della porta, dovuta alle capacità parassite. Nel paragrafo successivo verrà esaminata nel dettaglio una tecnica di ridimensionamento dei transistor nelle reti combinatorie CMOS complesse.

- **Dimensionamento progressivo dei transistor** Una tecnica alternativa al dimensionamento uniforme dei transistor appena citato (in cui le dimensioni di ciascun transistor sono riscalate di un fattore comune), è quella del dimensionamento progressivo (Figura 6-14). Con riferimento all'Eq. (6.3), vediamo che la resistenza equivalente a M_1 (R_1) appare N volte nella relazione del tempo di ritardo, la resistenza associata a M_2 (R_2) appare $N - 1$ volte, ecc. Appare chiaro da questa equazione che i transistor dovrebbero essere dimensionati in modo tale che la resistenza più piccola sia R_1 la successiva sia R_2 e così via. Conseguentemente, un dimensionamento progressivo dei transistor che porta ad un sostanziale beneficio è: $M_1 > M_2 > M_3 >... > M_N$. Questo approccio riduce le resistenze dominanti, pur contenendo l'incremento delle capacità entro certi limiti. Per una trattazione accurata del dimensionamento ottimo dei transistor in una lunga catena rinviamo a testi quali [Shoji88, pp. 131–143]. Bisogna però porre attenzione a una sottile, ma importante, insidia di questo metodo. Sebbene il dimensionamento progressivo dei transistor sia relativamente semplice sulla carta, non lo è in un layout reale: molto spesso le regole di layout impongono al progettista di collocare i transistor lontani l'uno dall'altro con il conseguente aumento delle capacità parassite. Ciò potrebbe vanificare ogni beneficio del dimensionamento progressivo.
- **Riordinare gli ingressi** Quando si ha a che fare con reti combinatorie complesse, alcuni dei segnali possono essere più critici di altri. Non tutti i segnali di ingresso arrivano nello stesso istante (a causa dei diversi tempi di propagazione dei blocchi logici precedenti, per esempio). Un segnale di ingresso a un porta logica è definito *critico* se è l'ultimo tra i segnali di ingresso ad assumere il valore logico stabile. Il percorso attraverso il circuito logico che determina la velocità effettiva di una certa struttura è detto *percorso o cammino critico*.

  Posizionare il transistor che corrisponde al cammino critico vicino all'uscita della porta logica può aumentarne la velocità di commutazione, come dimostrato nella Figura 6-15. Il segnale In_1 è quello critico. Supponiamo che In_2 e In_3 siano alti, che In_1 esegua una commutazione $0 \rightarrow 1$ e che la capacità C_L sia inizialmente carica al livello alto. Nel caso (a), non esiste alcun cammino conduttivo verso massa finché M_1 non è acceso, ma questo,

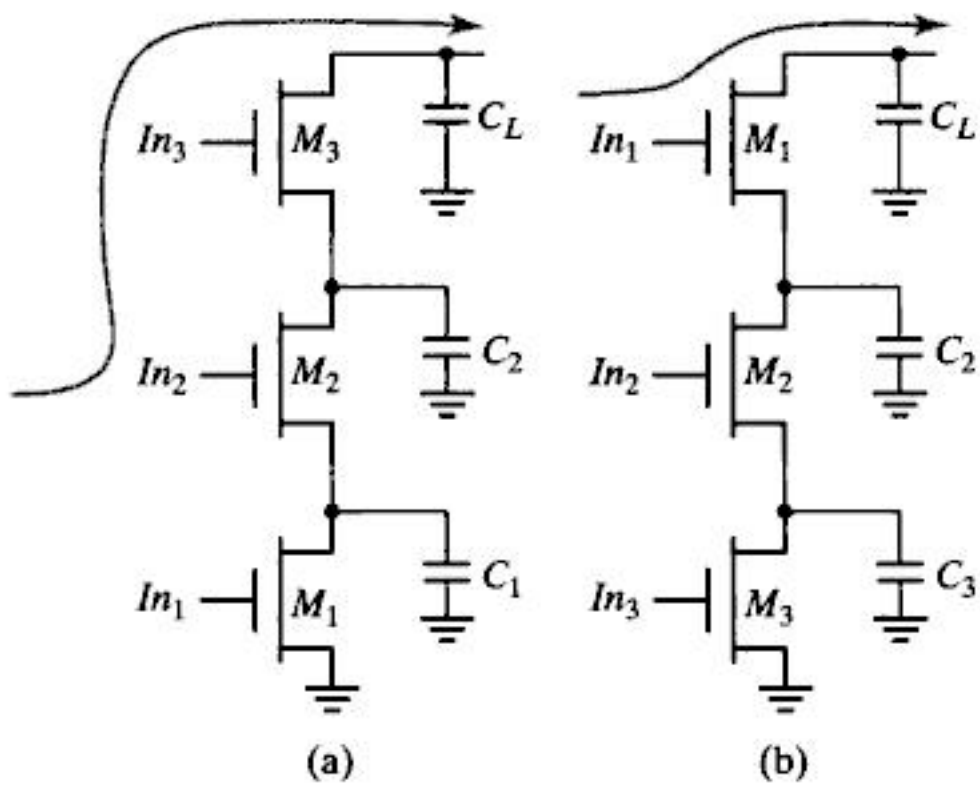


Figura 6-15 Effetto del riordinamento degli ingressi sul tempo di ritardo. Il segnale In_1 è l'ingresso critico.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

La rappresentazione del ritardo di una porta logica nella forma dell'Eq. (6.6) è una semplice relazione lineare, come illustrato graficamente nella Figura 6-18: il tempo di propagazione è riportato in funzione del fan-out per un invertitore e per una porta NAND a due ingressi. La pendenza della retta rappresenta il lavoro logico della porta, mentre la sua intercetta è il tempo di ritardo intrinseco. Il grafico mostra che possiamo regolare il tempo di ritardo sia attraverso il fan-out effettivo (dimensionando in modo opportuno i transistor), sia scegliendo una porta logica caratterizzata da un differente lavoro logico. Osserviamo inoltre che il fan-out e il lavoro logico contribuiscono al ritardo in modo simile. Possiamo definire il prodotto dei due $h = fg$ il *lavoro effettivo della porta.*

Il tempo di propagazione complessivo attraverso il cammino critico di una rete combinatoria può essere espresso come:

$$t_p = \sum_{j=1}^{N} t_{p,j} = t_{p0} \sum_{j=1}^{N} \left(p_j + \frac{f_j g_j}{\gamma} \right) \tag{6.7}$$

Per dimensionare in modo ottimale le porte logiche lungo un cammino in una rete combinatoria complessa, usiamo una procedura simile a quella introdotta nel Capitolo 5 per una catena di invertitori. Calcolando le $N - 1$ derivate parziali e uguagliandole a zero, troviamo che **ciascuno stadio dovrebbe sopportare lo stesso *lavoro effettivo***:

$$f_1 g_1 = f_2 g_2 = \ldots = f_N g_N \tag{6.8}$$

Il lavoro logico lungo un cammino di una rete combinatoria si calcola moltiplicando il lavoro logico di tutte le catene lungo il cammino, ottenendo il *lavoro logico globale* del percorso G:

$$G = \prod_{1}^{N} g_i \tag{6.9}$$

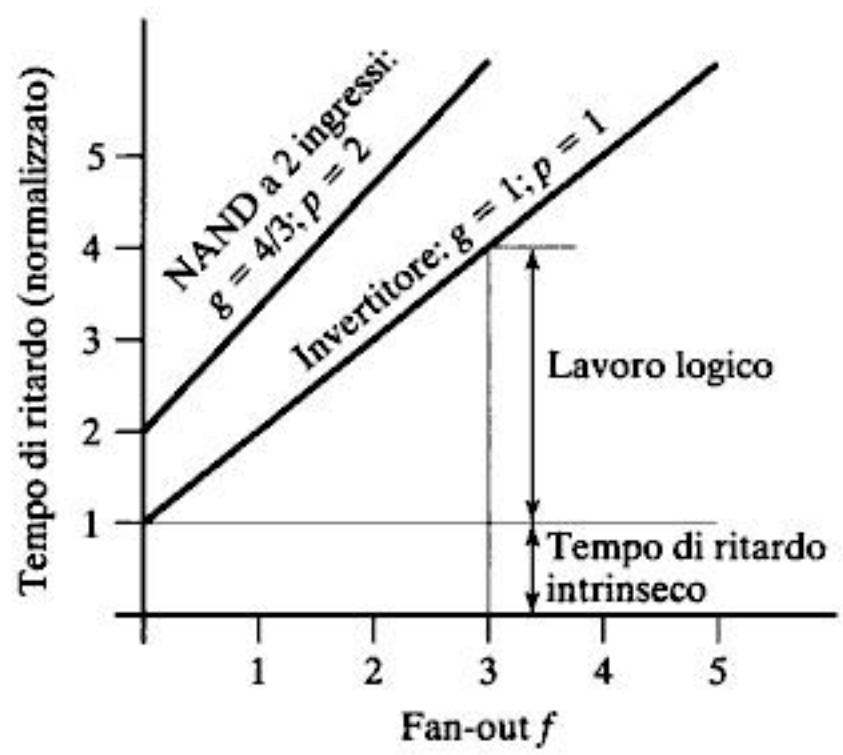


Figura 6-18 Tempo di ritardo di un invertitore e di una porta NAND a due ingressi in funzione del fan-out.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

$$\alpha_{0\to 1} = \frac{N_0}{2^N} \cdot \frac{N_1}{2^N} = \frac{N_0 \cdot (2^N - N_0)}{2^{2N}} \tag{6.20}$$

dove N_0 è il numero di configurazioni degli ingressi a cui corrisponde un valore 0 dell'uscita e N_1 è il numero di configurazioni di ingresso che corrispondono a un 1 in uscita. A scopo illustrativo, consideriamo la porta NOR la cui tabella di verità è mostrata nella Tabella 6-6. Assumiamo che durante un ciclo di clock sia possibile al più una commutazione degli ingressi e che gli ingressi della porta NOR abbiano una distribuzione di probabilità uniforme. In altre parole, le quattro possibili configurazioni degli ingressi A e B (00, 01, 10 e 11) sono equiprobabili.

Dalla Tabella 6-6 e dall'Eq. (6.20) la probabilità di commutazione dell'uscita della porta NOR a due ingressi può essere calcolata come:

$$\alpha_{0\to 1} = \frac{N_0 \cdot (2^N - N)}{2^{2N}} = \frac{3 \cdot (2^2 - 3)}{2^{2\cdot 2}} = \frac{3}{16} \tag{6.21}$$

Tabella 6-6 Tabella di verità di una porta NOR a due ingressi.

A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

Problema 6.3 Porta XOR a *N* ingressi

Calcolare il fattore di attività di una porta XOR a N ingressi nell'ipotesi che i segnali di ingresso siano equiprobabili e indipendenti.

Descrizione statistica dei segnali Il fattore di attività di una porta logica è funzione della descrizione statistica degli ingressi. Calcolare il fattore di attività assumendo una distribuzione uniforme non è una tecnica sufficientemente accurata, poiché la propagazione dei segnali attraverso le diverse porte logiche porta a sostanziali variazioni della descrizione statistica dei segnali lungo il cammino. Per fare un esempio, prendiamo in esame ancora una volta la porta NOR a due ingressi e siano p_a e p_b le probabilità che gli ingressi A e B assumano il valore logico *uno*. Ipotizziamo inoltre che non vi sia alcuna correlazione tra gli ingressi. La probabilità che il nodo di uscita sia 1 è data da:

$$p_1 = (1 - p_a)(1 - p_b) \tag{6.22}$$

Quindi, la probabilità di avere una transizione da 0 a 1 all'uscita è:

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Come realizzare un *pull-up* migliore

Modificando opportunamente l'architettura appena illustrata è possibile costruire porte logiche a rapporto in grado di eliminare completamente il consumo statico di potenza e con escursione logica in uscita pari alla tensione di alimentazione. Tali porte si fondano su due concetti chiave: la *logica differenziale* e la *retroazione positiva*. Una porta logica differenziale richiede che ciascun ingresso sia disponibile in formato complementare (sia *diretto* sia *negato*) e fornisce due uscite con valore logico complementare. La retroazione positiva assicura che il dispositivo di carico possa essere spento quando è necessario. Un esempio di questa famiglia logica, detta DCVSL *(Differential Cascode Voltage Switch Logic)*, è presentato nella Figura 6-30a [Heller84].

Le due reti di *pull-down* PDN1 e PDN2 sono realizzate con soli transistor NMOS e sono mutuamente esclusive (complementari); ciò significa che, quando la rete PDN1 è attiva, la rete complementare PDN2 è spenta e vice versa, cosicché vengono implementate simultaneamente sia la funzione logica richiesta sia quella inversa. Supponiamo inizialmente che le uscite *Out* e $\overline{Out}$ abbiano rispettivamente valore logico 1 e 0 e che, per una data configurazione degli ingressi, la rete PDN1 venga accesa e la rete PDN2 venga spenta. In queste condizioni $\overline{Out}$ si trova in uno stato di alta impedenza (isolato), poiché sia M_2 sia PDN2 sono interdetti. Sebbene vi sia ancora una connessione tra l'uscita *Out* e l'alimentazione grazie al transistor M_1, attivando la rete PDN1, la tensione del nodo *Out* diminuisce. Se PDN1 è abbastanza forte rispetto a M_1, la tensione del nodo *Out* scende sotto il valore $V_{DD} - |V_{Tp}|$, ossia il punto in cui M_2 si accende e inizia a caricare il nodo $\overline{Out}$ verso V_{DD}, spegnendo definitivamente M_1. Ciò, a sua volta, consente a PDN1 di scaricare completamente *Out* a massa. La

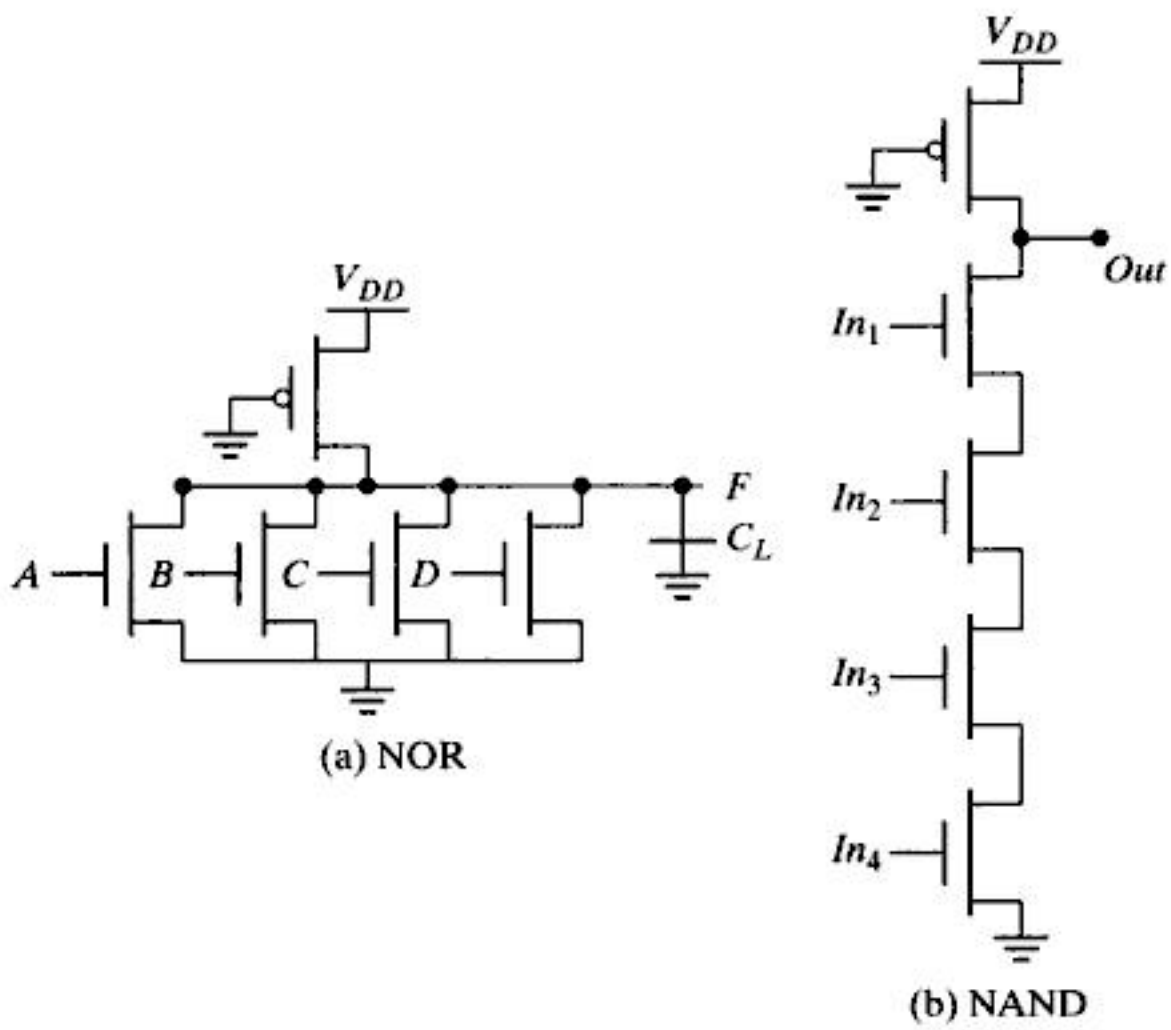


Figura 6-29 Porte NOR e NAND a quattro ingressi pseudo-NMOS.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

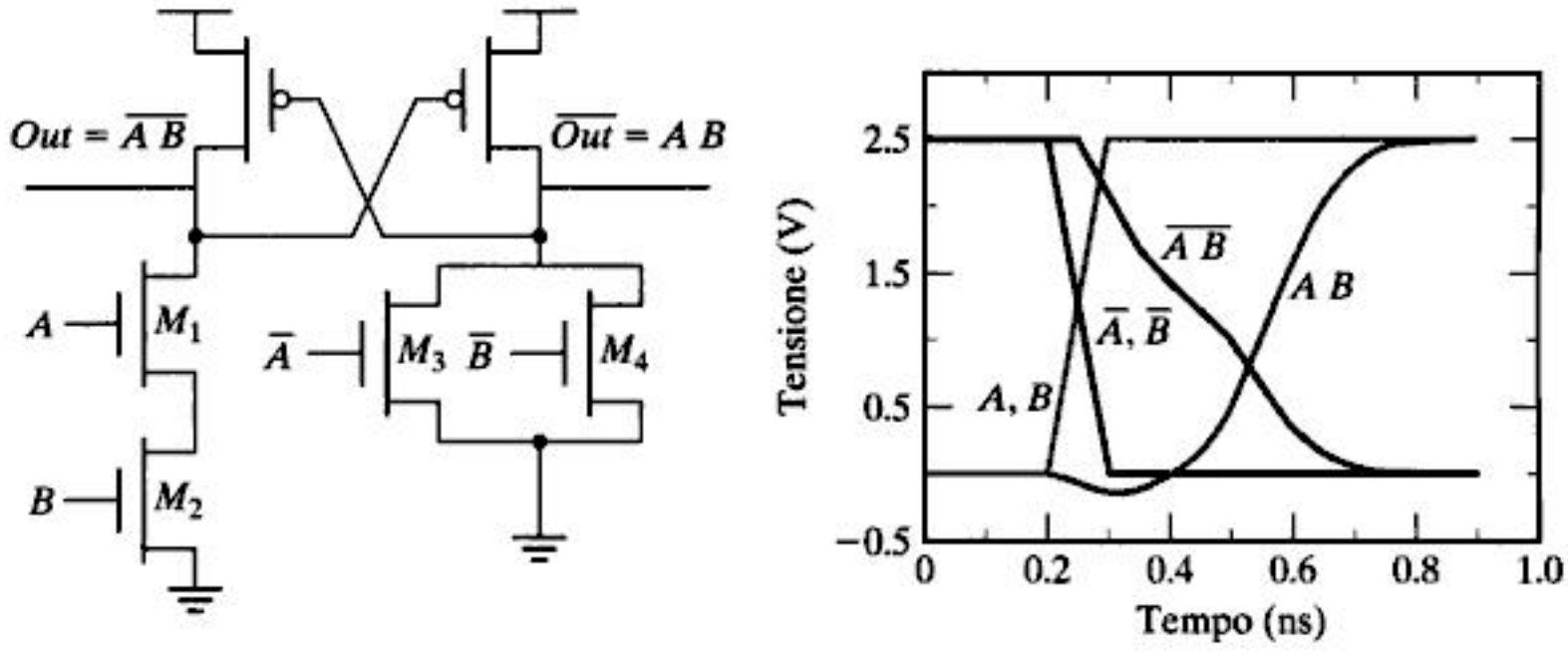


Figura 6-31 Risposta dinamica di una porta AND/NAND DCVSL. Le dimensioni di M_1 e M_2 sono 1 μm/0.25 μm, quelle di M_3 e M_4 sono 0.5 μm/0.25 μm e quelle dei due dispositivi PMOS sono 1.5 μm/0.25 μm.

Considerazioni progettuali: uscita singola e uscita differenziale

Le porte logiche DCVSL forniscono *uscite differenziali* (o *complementari*), ovvero sono disponibili simultaneamente sia il segnale diretto (V_{out}) sia il segnale negato ($\overline{V}_{out}$). Questo rappresenta un netto vantaggio, non essendo necessario l'impiego di un invertitore aggiuntivo per generare il segnale complementare. Abbiamo visto che l'implementazione differenziale di una funzione complessa può dimezzare il numero di porte logiche richieste, quindi anche il numero di porte presenti nel cammino critico del segnale può essere sensibilmente ridotto. Infine, questo approccio evita molti dei problemi di temporizzazione derivanti dall'inserimento di invertitori addizionali nei cammini di propagazione dei segnali. Spesso, in un circuito digitale è necessario disporre sia del segnale diretto sia del suo negato. Se il segnale complementare è generato da un invertitore, esso sarà ritardato rispetto al segnale originale (Figura 6-32a), causando problemi di temporizzazione, specialmente in circuiti ad alta velocità. Questo problema è parzialmente, se non completamente, risolto nelle famiglie logiche con uscita differenziale (Figura 6-32b).

Alla luce di tutte queste buone caratteristiche, perché non usare sempre una logica differenziale? Il motivo sta nel fatto che la natura differenziale della porta raddoppia di fatto il numero di connessioni

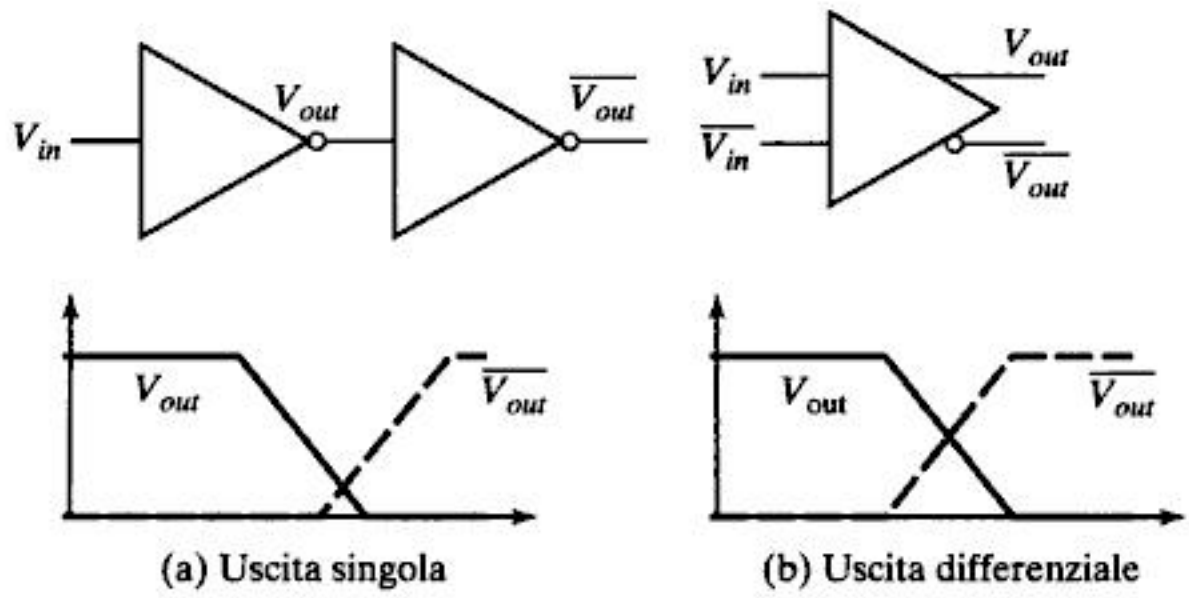


Figura 6-32 Confronto tra porta logica a uscita singola (a) e differenziale (b).

Materiale protetto da copyright

e di piste che devono essere impiegate, conducendo spesso ad un layout molto complesso e ingombrante. A questo si somma l'occupazione aggiuntiva di area dovuta alla presenza della funzione complementare. Infine anche il consumo dinamico di potenza risulta abbastanza elevato. ■

6.2.3 Logica pass-transistor

Concetti fondamentali

Un'alternativa alla logica CMOS complementare molto diffusa è la *logica pass-transistor*, che mira a ridurre il numero di dispositivi necessari all'implementazione di una data funzione logica collegando gli ingressi primari sia ai terminali di gate sia ai terminali di source/drain [Radhakrishnan85]. Tale approccio è in netto contrasto con le famiglie logiche viste finora, nelle quali gli ingressi primari facevano capo solo ai terminali di gate dei MOSFET.

La Figura 6-33 mostra l'implementazione di una porta AND realizzata proprio secondo questa tecnica. Quando l'ingresso B è alto, il transistor superiore è acceso e il segnale A viene riportato all'uscita F. Invece, quando B è basso, il transistor inferiore è acceso e all'uscita passa lo 0. A prima vista, il dispositivo pilotato da $\overline{B}$ sembra ridondante, ma la sua presenza è essenziale per assicurare che la porta sia statica, cioè garantire che in qualunque condizione vi sia sempre un cammino a bassa resistenza che connetteta l'uscita o alla massa o alla tensione di alimentazione.

Il vantaggio di questo tipo di logica è che sono necessari pochi transistor per implementare una data funzione. L'implementazione della porta AND della Figura 6-33, per esempio, ne richiede solo quattro (inclusi quelli necessari a invertire il segnale B), mentre l'implementazione complementare ne richiederebbe sei. Il minor numero di dispositivi ha come conseguenza anche la riduzione delle capacità parassite.

Sfortunatamente, come abbiamo già discusso in precedenza, un dispositivo NMOS è molto efficiente nel trasmettere uno 0, ma trasmette male il valore logico 1: quando il pass-transistor è impiegato per trasmettere V_{DD}, la tensione di uscita raggiunge al massimo il valore $V_{DD} - V_{Tn}$. In realtà, la situazione è ulteriormente peggiorata dall'effetto body, a causa dell'elevata tensione presente tra il source e il substrato in queste condizioni. Consideriamo il caso in cui il pass-transistor sta caricando un nodo con i terminali di gate e di drain mantenuti a V_{DD}. Detto x il nodo corrispondente al terminale di source del pass-transistor NMOS, x sarà caricato fino alla tensione $V_{DD} - V_{Tn}(V_x)$, ossia:

$$V_x = V_{DD} - [V_{Tn0} + \gamma(\sqrt{|2\phi_f| + V_x} - \sqrt{|2\phi_f|})] \tag{6.30}$$

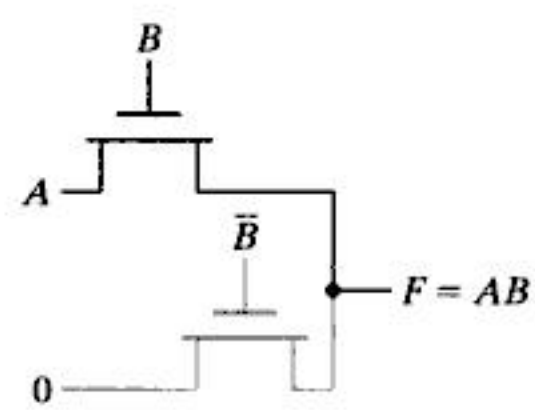


Figura 6-33 Implementazione di una porta AND con pass-transistor.

Materiale protetto da copyright

Esempio 6.9 Escursione logica di un circuito a pass-transistor

La risposta dinamica riportata nella Figura 6-34 mostra un NMOS che sta caricando una capacità. Il terminale di drain è mantenuto a V_{DD} e al gate viene applicata una rampa di tensione da 0 V a V_{DD}. Supponendo che il nodo x sia inizialmente a 0 V, osserviamo che inizialmente l'uscita viene caricata molto velocemente, ma in seguito il transitorio prosegue molto più lento. Infatti, la tensione gate-source (e quindi la conducibilità) del transistor diminuisce via via che la tensione del nodo di uscita x si avvicina a $V_{DD} - V_{Tn}$ riducendo drasticamente l'intensità della corrente disponibile per la carica del nodo. Il calcolo manuale dell'Eq. (6.30) fornisce un valore di tensione di 1.8 V, molto vicino al valore simulato.

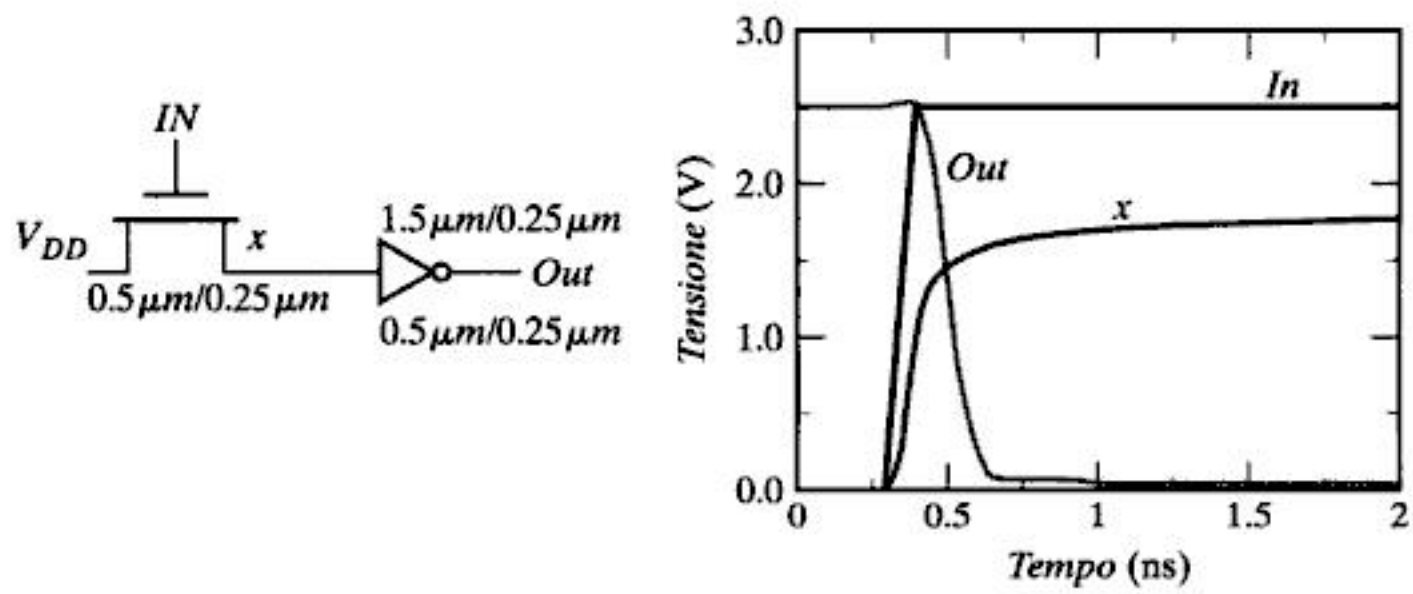


Figura 6-34 Risposta dinamica di un pass-transistor NMOS. Notare il rallentamento nel processo di carica dopo la risposta iniziale molto veloce (V_{DD} = 2.5 V).

ATTENZIONE: L'esempio precedente dimostra che **le porte logiche a pass-transistor non possono essere messe in cascata connettendo l'uscita di una porta all'ingresso della successiva, qualora tale ingresso sia il terminale di gate di un transistor**. Ciò è illustrato nella Figura 6-35a, dove l'uscita di M_1 (nodo x) pilota il gate di un altro dispositivo MOS. Il nodo x può essere caricato fino a $V_{DD} - V_{Tn1}$. Se il nodo C ha un'escursione completa tra alimentazione e massa, il nodo Y si caricherà al massimo fino alla tensione del nodo x *meno la tensione di soglia* V_{Tn2} *del dispositivo* M_2, ossia $V_{DD} - V_{Tn1} - V_{Tn2}$. La Figura 6-35b, invece, mostra il transistor M_1 che pilota il drain (source) di M_2 e vi è una caduta di tensione pari a una sola volta la tensione di soglia del transistor. Questo è il modo corretto di mettere in cascata le porte logiche a pass-transistor.

Esempio 6.10 VTC di una porta AND a pass-transistor

La caratteristica di trasferimento ingresso-uscita di una porta logica a pass-transistor mostra poca somiglianza con quella di una porta logica CMOS complementare. Consideriamo la porta AND disegnata nella Figura 6-36: analogamente alla logica CMOS complementare, la VTC della porta a pass-transistor è dipendente dalla configurazione

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

degli invertitori CMOS statici per rigenerare il segnale. Inserendo un invertitore all'uscita della porta a pass-transistor, la VTC assomiglia a quella di una porta CMOS complementare.

L'implementazione a pass-transistor richiede anche una minore energia per far commutare un nodo grazie alla escursione ridotta del segnale di uscita. Consideriamo il circuito a pass-transistor in Figura 6-34, assumiamo che la tensione di drain sia V_{DD}, che il nodo x sia inizialmente a 0 V e che il gate esegua una commutazione verso V_{DD}. La tensione del nodo x sale da 0 V a $V_{DD} - V_{Tn}$ e l'energia sottratta all'alimentazione per caricare il nodo di uscita del pass-transistor è data da

$$\begin{aligned} E_{0\to 1} &= \int_0^1 P(t)dt = V_{DD}\int_0^1 i_{DD}(t)dt = \\ &= V_{DD}\int_0^{(V_{DD}-V_{Tn})} C_L dV_{out} = C_L \cdot V_{DD} \cdot (V_{DD} - V_{Tn}) \end{aligned} \tag{6.31}$$

Nonostante il minor consumo dinamico di potenza, il circuito consuma potenza in condizioni statiche quando il nodo x è alto. Infatti, l'escursione ridotta del nodo x potrebbe essere insufficiente per spegnere completamente il PMOS dell'invertitore CMOS che segue in cascata.

Logica pass-transistor differenziale

Nei sistemi ad alta velocità è spesso impiegata una famiglia logica complementare basata su pass-transistor, detta *CPL (Complementary Pass-Transistor)* o *DPL (Differential Pass-Transistor)*. L'idea di fondo (simile alla logica DCVSL) è di poter disporre degli ingressi sia diretti sia complementari e di produrre in uscita sia il valore logico diretto sia quello negato. Alcune porte logiche CPL elementari (AND/NAND, OR/NOR e XOR/NXOR) sono illustrate nella Figura 6-37. Queste porte godono di alcune interessanti proprietà:

- Essendo il circuito costituito da porte *differenziali*, i valori complementari per gli ingressi e le uscite sono sempre disponibili. Sebbene sia necessaria una circuiteria aggiuntiva per la generazione dei segnali complementari da quelli primari, l'architettura differenziale ha il vantaggio che alcune porte più complesse come la XOR e i sommatori possono essere realizzati molto più agevolmente con un numero ridotto di transistor. Inoltre, la possibilità di disporre degli ingressi diretti e negati elimina la necessità di usare invertitori, come invece accade spesso nella logica CMOS complementare e pseudo-NMOS.
- La logica CPL appartiene alla classe delle porte *statiche*, poiché i nodi di uscita sono sempre connessi ora a massa ora a V_{DD} attraverso un cammino a bassa resistenza. Ciò è molto importante per rimediare a eventuali disturbi.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

le due porte, altrimenti potrebbero sorgere problemi di conflitto tra i dispositivi dei *level restorer* delle due porte, compromettendo le prestazioni e il funzionamento del circuito.

Soluzione 2: pass-transistor con tensione di soglia multipla. Una soluzione di carattere tecnologico al problema della degradazione del valore logico di uscita, causato dalla tensione di soglia in logica pass-transistor, è quella di impiegare dispositivi con differente tensione di soglia. L'uso di dispositivi con tensione di soglia nulla per realizzare i pass-transistor NMOS elimina gran parte della caduta di tensione e permette di trasmettere un valore prossimo a V_{DD}. Tutti gli altri dispositivi usati al di fuori della rete di pass-transistor (cioè, gli invertitori) sono implementati con dispositivi standard (tensione di soglia positiva). L'impiego di transistor con tensione di soglia multipla sta diventando sempre più comune e comporta solo modeste variazioni agli attuali passi dei processi di fabbricazione. Tuttavia bisogna considerare che, anche se il livello di drogaggio fosse calibrato in modo da ottenere una tensione di soglia nulla, l'effetto body, sempre presente, impedirebbe di ottenere un'escursione logica piena fino a V_{DD}.

Usare pass-transistor con tensione di soglia nulla deteriorerebbe inoltre caratteristiche quali il consumo di potenza a causa delle elevate correnti di sottosoglia anche quando V_{GS} è inferiore a V_T. La Figura 6-43 dimostra quanto appena accennato, evidenziando un possibile cammino diretto tra l'alimentazione e la massa. Sebbene questi percorsi conduttivi non siano critici al funzionamento del circuito, quando i dispositivi sono fatti commutare di continuo, essi costituiscono un costo aggiuntivo in termini di energia quando il circuito si trova in condizioni stazionarie.

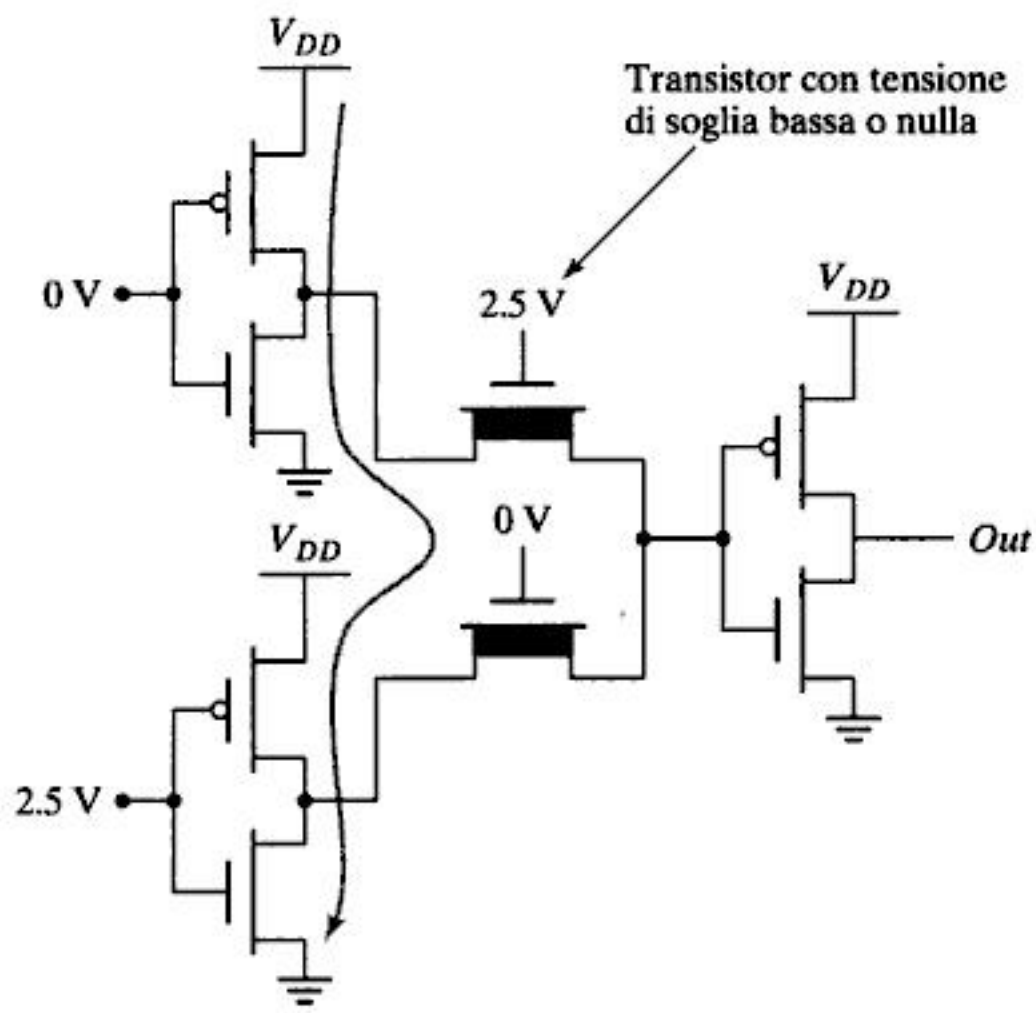
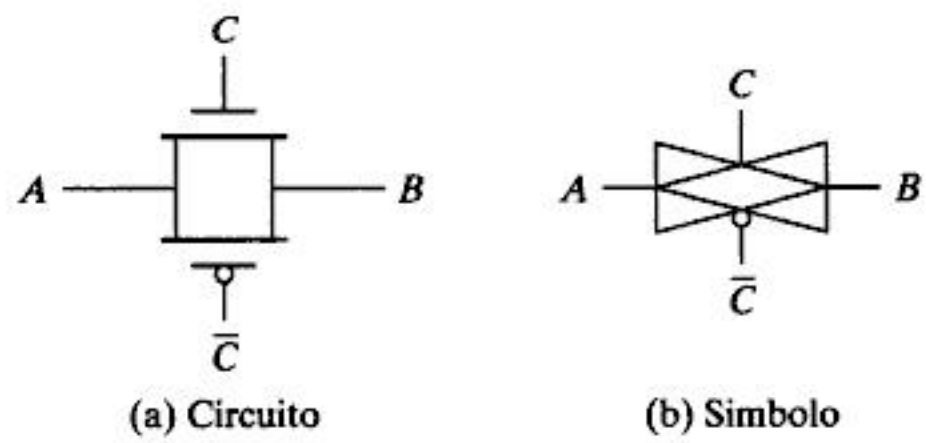


Figura 6-43 Consumo statico di potenza nell'utilizzare pass-transistor con tensione di soglia nulla.

Materiale protetto da copyright



(a) Circuito (b) Simbolo

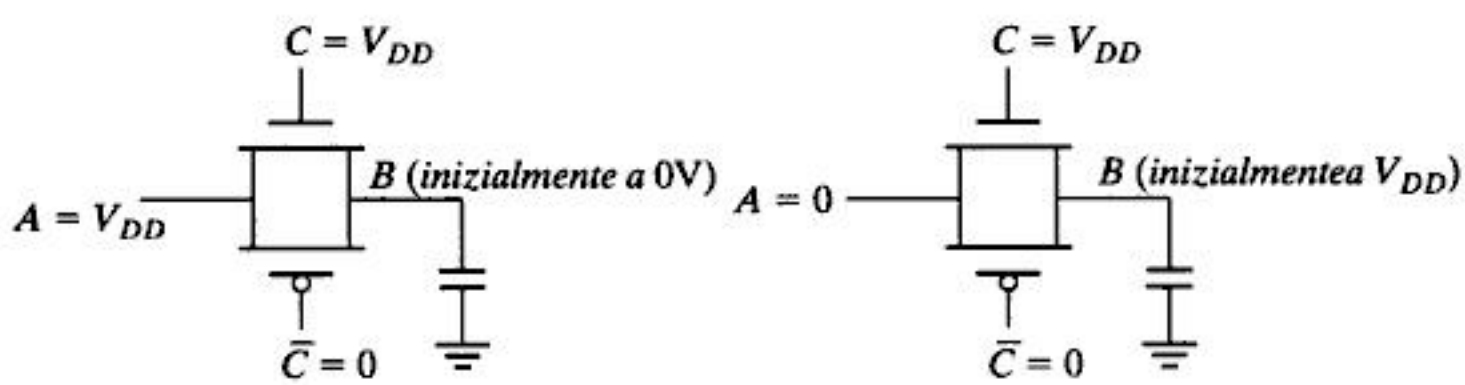
Figura 6-44 Gate di trasmissione CMOS.

Soluzione 3: logica a gate di trasmissione La soluzione più diffusa per affrontare il problema della caduta di tensione è l'uso di *gate di trasmissione (transmission gate logic)*.[4] Questa tecnica si fonda sulle proprietà complementari dei transistor NMOS e PMOS: i dispositivi NMOS trasmettono bene gli 0 logici e male gli 1; invece, i dispositivi PMOS trasmettono correttamente gli 1 logici, ma male gli 0. Come abbiamo già visto in occasione della logica CMOS complementare, l'approccio ideale è quello di usare un transistor NMOS per scaricare il nodo di uscita e un transistor PMOS per caricarlo. Il gate di trasmissione sintetizza in un unico componente le proprietà migliori di entrambi i transistor, combinando un NMOS e un PMOS in parallelo come illustrato nella Figura 6-44a. I segnali di controllo del gate di trasmissione (C e $\overline{C}$) sono complementari. Il gate di trasmissione si comporta come un interruttore bidirezionale controllato dal segnale C. Quando $C = 1$, entrambi i MOSFET sono attivi, permettendo al segnale di passare attraverso il gate di trasmissione. In breve,

$$A = B \quad \text{se} \quad C = 1 \tag{6.32}$$

Al contrario, con $C = 0$, entrambi i transistor sono interdetti e il gate di trasmissione diventa un circuito aperto tra A e B. La Figura 6-44b riporta il simbolo convenzionalmente usato per il gate di trasmissione.

Consideriamo il processo di carica del nodo B da 0 a V_{DD} attraverso il gate di trasmissione in Figura 6-45a. Il nodo A è mantenuto a V_{DD} e il gate di trasmissione è abili-



(a) Carica del nodo B a V_{DD} (b) Scarica del nodo B a 0V

Figura 6-45 Il gate di trasmissione fornisce un'escursione logica completa tra massa e alimentazione.

[4]Il gate di trasmissione è solo una delle possibili soluzioni. Sono state proposte anche altre tecniche per costruire reti che combinano insieme NMOS e PMOS: un esempio di queste è la logica a doppio pass transistor (DPL, Double Pass-Transistor) [Bernstein98, pp. 84].

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

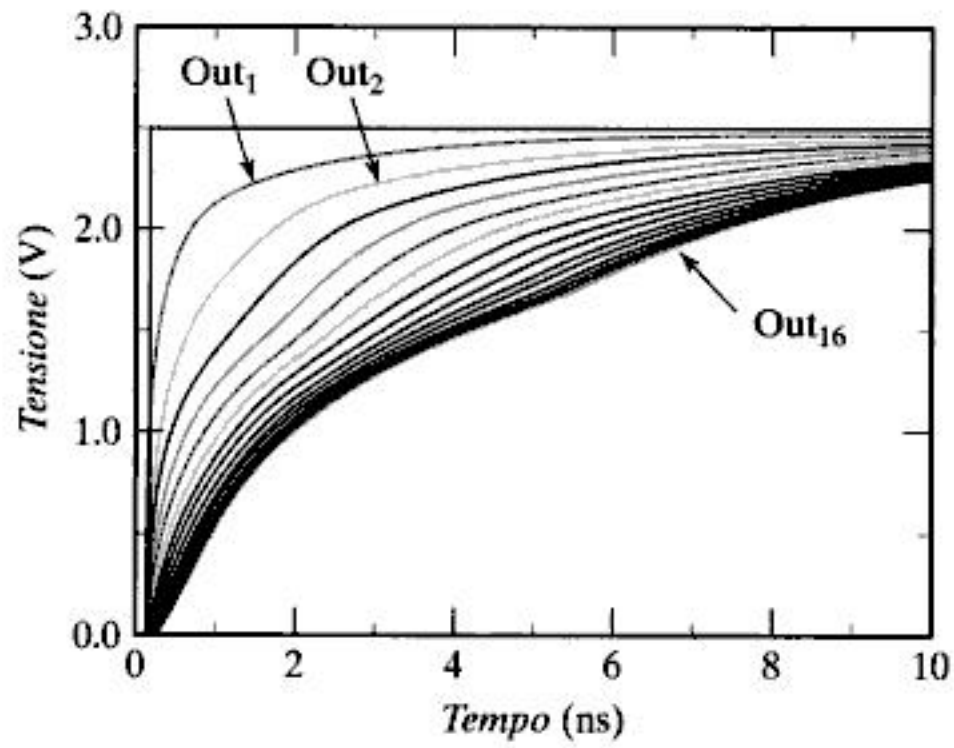


Figura 6-50 Tempo di propagazione di una catena di gate di trasmissione.

Il modo più comune per affrontare il problema dei lunghi tempi di propagazione delle catene di pass-transistor (gate di trasmissione) è inserire dei buffer ogni m stadi per rigenerare il segnale (Figura 6-51). Detto t_{buf} il tempo di propagazione per ciascun buffer, il tempo di propagazione complessivo lungo la catena di gate di trasmissione e buffer è calcolato come segue:

$$\begin{aligned} t_p &= 0{,}69\left[\frac{n}{m}CR_{eq}\frac{m(m+1)}{2}\right]+\left(\frac{n}{m}-1\right)t_{buf} \\ &= 0{,}69\left[CR_{eq}\frac{n(m+1)}{2}\right]+\left(\frac{n}{m}-1\right)t_{buf} \end{aligned} \tag{6.38}$$

Il tempo di ritardo ottenuto mostra una dipendenza lineare con il numero n di interruttori nella catena, contrariamente al caso senza buffer visto sopra, nel quale la dipendenza era quadratica. Il numero ottimo m_{opt} di gate di trasmissione tra due buffer consecutivi può essere calcolato facilmente derivando t_p rispetto a m e uguagliando la derivata a 0:

$$\frac{\partial t_p}{\partial m}=0 \qquad m_{opt} = 1{,}7\sqrt{\frac{t_{buf}}{CR_{eq}}} \tag{6.39}$$

Ovviamente il numero di interruttori per ciascun segmento della catena aumenta al crescere di t_{buf}. Nelle attuali tecnologie, m_{opt} è tipicamente 3 o 4. Questa semplice analisi ignora tuttavia che t_{buf} è a sua volta dipendente dal carico e quindi da m. Uno studio più accurato, che tenga conto anche di questo fattore, sarà presentato nel Capitolo 9.

Esempio 6.14 Catena di gate di trasmissione

Consideriamo la stessa catena di 16 gate di trasmissione dell'esempio precedente. I buffer mostrati nella Figura 6-51 possono essere semplici invertitori (invece di coppie

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

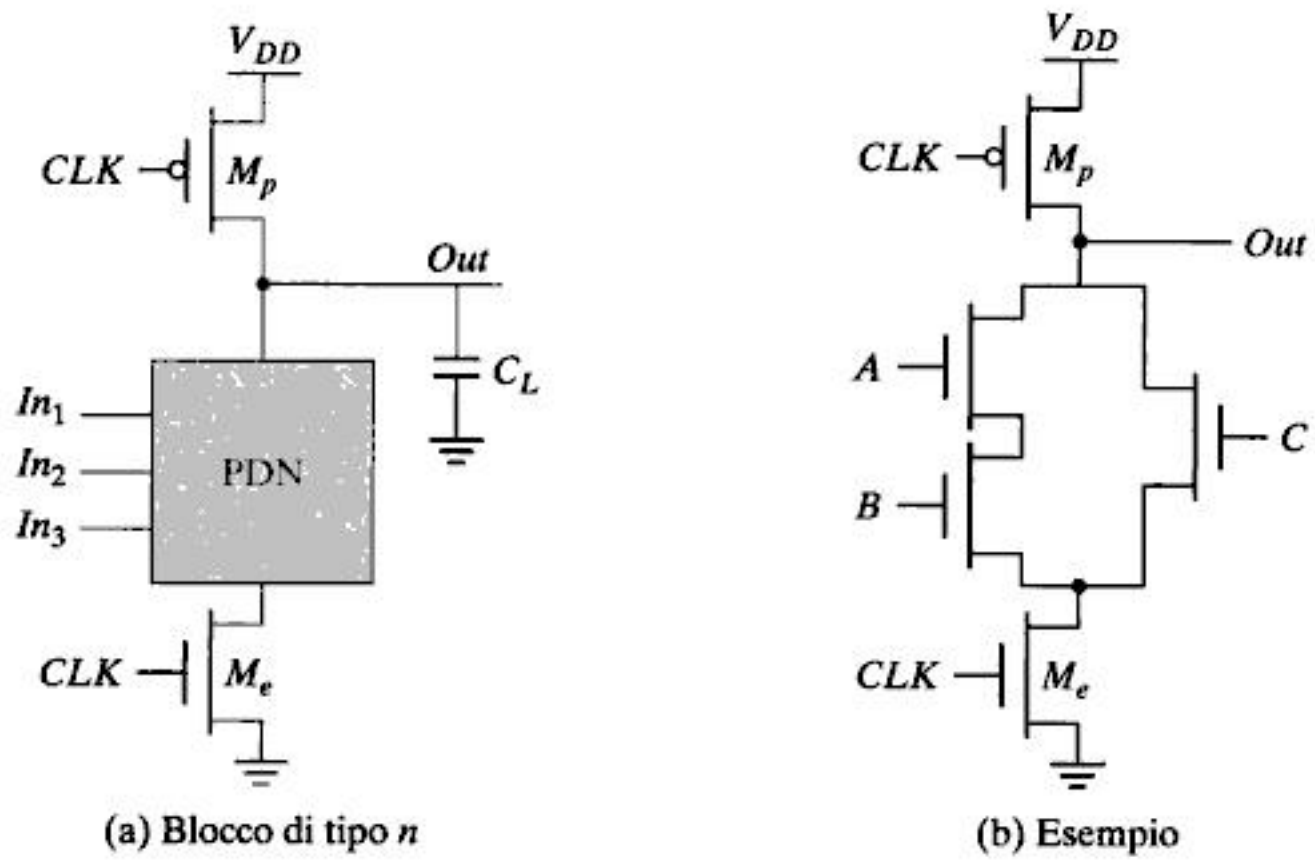


Figura 6-52 Struttura di principio di una porta logica dinamica.

6.3.1 Principi base

Lo schema di principio di una porta logica dinamica di tipo n è illustrato nella Figura 6-52a. La rete di *pull-down*, PDN, è costruita esattamente come quella di una porta logica CMOS complementare. Il funzionamento di questo circuito è diviso in due parti fasi fondamentali: *precarica* e *valutazione*. L'alternanza di queste due fasi è sincronizzata dal *segnale di clock CLK*.

Precarica

Quando $CLK = 0$, il nodo di uscita è precaricato a V_{DD} dal transistor PMOS M_p. Durante questa fase il transistor NMOS di valutazione M_e è spento, in modo tale che il *pull-down* sia isolato dalla massa. M_e elimina quindi ogni possibile consumo statico durante la fase di precarica (cioè, dovuto alle eventuali correnti statiche che possono fluire dall'alimentazione alla massa, se il *pull-down* e il transistor di precarica fossero contemporaneamente accesi).

Valutazione

Per $CLK = 1$, il transistor M_p è spento e il transistor di valutazione M_e acceso. L'uscita può essere scaricata a massa in base alla configurazione degli ingressi e alla topologia del *pull-down*. Se la configurazione di ingresso è tale che la rete di *pull-down* sia accesa, allora è presente un percorso conduttivo a bassa resistenza tra la massa e il nodo di uscita che viene scaricato a 0 V. Invece, se il PDN è spento, la capacità C_L (che è la somma delle capacità dei drain dei transistor della porta, delle capacità parassite delle piste di interconnessioni e la capacità di ingresso dello stadio successivo) rimane carica al valore impostato nella precedente fase di precarica. Durante la fase di valutazione, il solo percorso conduttivo possibile è dall'uscita verso la linea di massa, quindi, una volta scaricato il nodo *Out* non può essere caricato nuovamente fino all'inizio della successiva fase di precarica. **Gli ingressi della porta logica possono dunque eseguire al più una commutazione durante la fase di valu-**

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

immunità al rumore e la tensione di soglia logica sono metriche statiche e non sono funzione del tempo. Per essere completamente funzionale, una porta dinamica passa attraverso successive fasi di precarica e valutazione. Un analisi puramente statica di tale porta non avrebbe senso. Durante la fase di valutazione, il *pull-down* di un invertitore dinamico inizia a condurre quando la tensione del segnale di ingresso supera la tensione di soglia (V_{Tn}) del transistor NMOS. Quindi, è ragionevole assumere che la tensione di soglia logica (V_M) così come i livelli di tensione V_{IH} e V_{IL} siano uguali a V_{Tn}. Ciò si traduce in un margine di rumore NM_L molto basso.

Implementazione di porte logiche dinamiche duali

Una porta logica dinamica può anche essere implementata seguendo l'approccio duale, dove il nodo di uscita è connesso a un dispositivo di precarica NMOS, che carica l'uscita a 0 V, mentre la rete di *pull-down* è sostituita da un *pull-up* (PUN) realizzato con dispositivi PMOS. Il funzionamento è del tutto simile: nella fase di precarica, l'uscita è scaricata verso massa; durante la fase di valutazione l'uscita è caricata a V_{DD} solo se il PUN è attivo. La porta logica dinamica di tipo *p* ha lo svantaggio di essere più lenta della corrispondente di tipo *n* a causa della minore conducibilità dei transistor PMOS. ■

6.3.2 Velocità e consumo di potenza delle porte logiche dinamiche

I maggiori vantaggi della logica dinamica sono la velocità e la minore occupazione di area: infatti, il minor numero di dispositivi necessari all'implementazione di una data funzione logica è accompagnato anche dalla riduzione delle capacità di carico. Un'analisi del funzionamento di queste porte evidenzia alcune interessanti peculiarità. Dopo la fase di precarica l'uscita è alta. Se il segnale di ingresso ha un valore basso, non avviene alcuna commutazione dell'uscita e quindi il corrispondente tempo di propagazione, t_{pLH}, *è nullo*. La transizione alto-basso, d'altra parte, richiede che venga scaricata la capacità del nodo di uscita attraverso la rete di *pull-down*. Quindi, t_{pHL} è proporzionale a C_L e alla conducibilità della rete di *pull-down*. La presenza del transistor di valutazione rallenta la porta logica, poiché rappresenta una resistenza serie addizionale. L'eliminazione di questo transistor, sebbene non impedisca il corretto funzionamento del circuito, introdurrebbe un consumo statico di potenza e ne degraderebbe le prestazioni.

Le considerazioni appena fatte non sono completamente corrette, perché ignorano totalmente il tempo necessario alla precarica sulla velocità di commutazione della porta logica. Il tempo di precarica è determinato dalla velocità con cui il transistor PMOS carica la capacità C_L. In questo intervallo di tempo la rete di NMOS che realizza la funzione logica non può essere utilizzata. Molto spesso è possibile organizzare l'intero sistema, a un livello di astrazione più alto, in modo tale che la fase di precarica coincida temporalmente con altre operazioni del sistema che non coinvolgono la rete logica in questione. Per esempio, in un microprocessore la fase di precarica dell'unità logico-aritmetica potrebbe coincidere con la fase di decodifica dell'istruzione. I progettisti devono sempre tenere in considerazione la presenza di questo *tempo morto* nel funzionamento della logica dinamica, in modo da poterlo

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

bilità che l'uscita sia scaricata in valutazione è del 25%), mentre per la realizzazione statica si otterrebbe ancora 3/16. Sebbene questo esempio dimostri come il fattore di attività di una porta dinamica sia sempre maggiore della corrispondente porta logica statica, nell'analisi del consumo di potenza bisogna tenere presente che generalmente la capacità di carico fisicamente connessa all'uscita è minore.

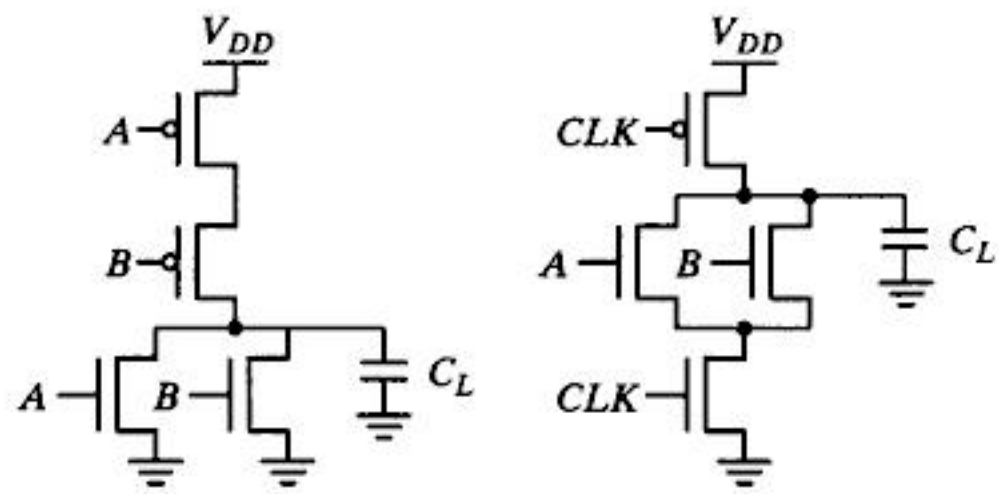


Figura 6-55 Porta NOR statica e porta NOR dinamica con rete di *pull-down* (blocco *n*).

Problema 6.8 Calcolo del fattore di attività

Calcolare il fattore di attività di una porta NAND a quattro ingressi in logica dinamica a precarica a V_{DD}, assumendo che gli ingressi siano indipendenti e siano a 1 con probabilità $p_{A=1} = 0.2$, $p_{B=1} = 0.3$, $p_{C=1} = 0.5$ e $p_{D=1} = 0.4$.

6.3.3 Robustezza e integrità del segnale nella logica dinamica

La logica dinamica rappresenta una soluzione alternativa alla logica statica quando si vogliono realizzare circuiti ad elevate prestazioni. Tuttavia, ci sono alcuni aspetti molto importanti che devono essere tenuti in considerazione, affinché un circuito dinamico funzioni correttamente. Tra questi, le correnti di perdita, il fenomeno di ridistribuzione della carica (*charge sharing*), gli accoppiamenti capacitivi e, soprattutto, l'accoppiamento capacitivo tra l'uscita e il segnale di clock. Questi problemi sono discussi in dettaglio nei paragrafi seguenti.

Correnti di perdita

Il funzionamento di un circuito dinamico si basa sull'immagazzinamento di carica elettrica nei nodi di un circuito. Questa carica è immagazzinata in modo dinamico, ossia accumulata nelle capacità associate al nodo stesso che poi è lasciato flottante. Idealmente, se il *pull-down* della funzione è spento, il nodo di uscita dovrebbe mantenere la tensione V_{DD} per l'intera durata della fase di valutazione. Tuttavia, le correnti di perdita presenti nel circuito tendono a scaricare gradualmente la capacità, originando potenziali malfunzionamenti della porta. La Figura 6-56a riassume le sorgenti di correnti di perdita in un semplice invertitore dinamico.

I contribuiti 1 e 2 rappresentano rispettivamente la *corrente inversa del diodo substrato-drain* e *la corrente di sottosoglia* del transistor NMOS del *pull-down* M_1. La carica

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Possiamo determinare quale di questi due casi sia quello esatto dal rapporto tra le capacità (C_a / C_L). La condizione limite tra i due casi può essere trovata sostituendo ΔV_{out} con V_{Tn} nel'Eq. (6.44), ottenendo:

$$\frac{C_a}{C_L} = \frac{V_{Tn}}{V_{DD} - V_{Tn}} \tag{6.45}$$

Il caso 1 si verifica quando il rapporto (C_a/C_L) è minore della quantità definita dall'Eq. (6.45). Altrimenti, è valida l'Eq. (6.44). In generale è consigliabile mantenere ΔV_{out} inferiore a $|V_{Tp}|$. Infatti, capita spesso che l'uscita della porta dinamica piloti un invertitore statico, nel qual caso valori più bassi di V_{out} potrebbero causare consumo statico di potenza. Infine, se la tensione di uscita scendesse addirittura sotto la soglia di commutazione dello stadio successivo, vi potrebbero essere problemi ancora più gravi, fino al malfunzionamento del circuito.

Esempio 6.18 Charge Sharing

Consideriamo l'impatto della ridistribuzione di carica nella porta logica della Figura 6-60 che implementa la funzione logica XOR: $y = A \oplus B \oplus C$. Per prima cosa bisogna identificare quali siano i nodi interni che causano la maggiore variazione della tensione di y. Per semplicità trascuriamo l'invertitore di carico e ipotizziamo che tutti gli ingressi siano bassi durante la fase di precarica e che le tensioni di tutti i nodi interni (V_a, V_b, V_c e V_d) siano inizialmente a 0 V.

La tabella di verità ci dice che l'uscita della funzione logica è bassa in 4 casi su 8. La massima variazione di tensione al nodo di uscita si ha quando è massima la capacità dei nodi interni che dà luogo al fenomeno di ridistribuzione della carica. Ciò

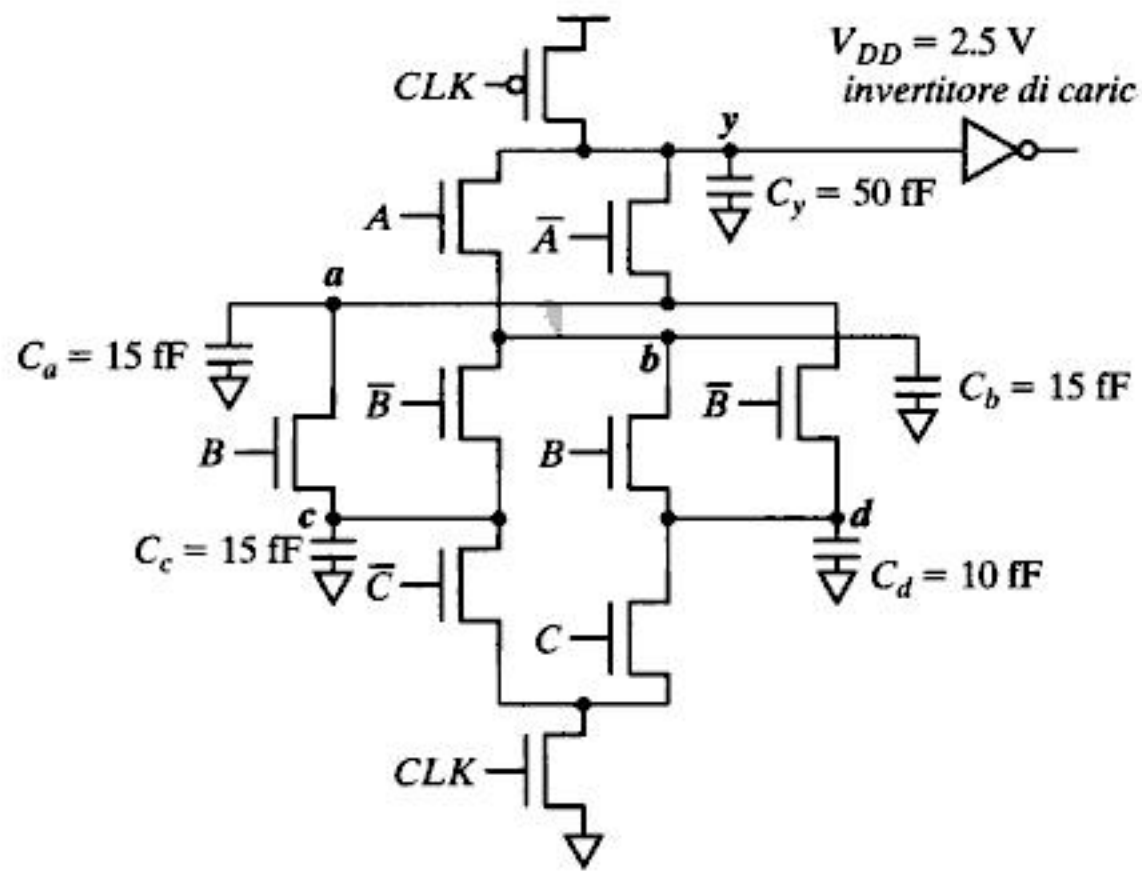


Figura 6-60 Esempio che mette in luce il problema della ridistribuzione della carica elettrica in un circuito dinamico.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

logica mettendo direttamente in cascata porte logiche dinamiche non funziona. Il problema viene illustrato meglio se consideriamo il circuito formato dalla cascata di due invertitori dinamici di tipo *n* (con rete di PDN) mostrato nella Figura 6-63a. Durante la fase di precarica ($CLK = 0$), le uscite di entrambi gli invertitori sono caricate a V_{DD}. Se l'ingresso primario *In* esegue la transizione $0 \rightarrow 1$ (Figura 6-63b), in corrispondenza del fronte di salita del clock l'uscita Out_1 inizia a scaricarsi. L'uscita del secondo invertitore dovrebbe rimanere nello stato di precarica (alla tensione V_{DD}) dato che il valore logico atteso è 1 (in valutazione Out_1 va a 0). Tuttavia, il tempo di propagazione attraverso il primo invertitore non è nullo e Out_1 viene scaricato a massa con un certo ritardo rispetto all'inizio della fase di valutazione e anche Out_2 inizia a scaricarsi. Fino a quando la tensione del nodo Out_1 è superiore alla tensione di soglia logica del secondo invertitore (approssimativamente uguale a V_{Tn}), esiste un cammino conduttivo tra Out_2 e la massa, e la carica immagazzinata nella capacità associata a Out_2 viene parzialmente persa. Tale cammino conduttivo si spegne solo dopo che la tensione del nodo Out_1 raggiunge V_{Tn}, spegnendo il transistor NMOS del *pull-down*. A causa di ciò, il nodo Out_2 viene lasciato a un livello di tensione intermedio. Il valore corretto non può essere ripristinato, poiché il funzionamento della logica dinamica risiede nella carica immagazzinata nella capacità di uscita, contrariamente alla logica statica in cui l'uscita è sempre a bassa impedenza e il dato viene rigenerato. La perdita di carica dalla capacità di uscita porta sia alla riduzione dei margini di immunità al rumore sia a possibili malfunzionamenti.

Il problema nasce dal fatto che le uscite di ciascuna porta (quindi l'ingresso della successiva) sono precaricati a 1 (i *pull-down* sono tutti accesi). Questo causa la scarica indesiderata dei nodi di uscita all'inizio della fase di valutazione. La soluzione al problema consiste nel porre tutti gli ingressi a 0 durante la fase di precarica. Procedendo in questa maniera, tutti i transistor nella rete di *pull-down* sono spenti alla fine della fase di precarica e le capacità di uscita non possono essere in alcun modo scaricate all'inizio della fase di valutazione. In altre parole, il corretto funzionamento è garantito se **gli ingressi possono compiere al più una commutazione $0 \rightarrow 1$ durante la fase di valutazione.**[5] I transistor sono accesi solo se necessario e, comunque, al più una volta per ciclo di clock. Sono state proposte un certo numero tecniche di progettazione che soddisfano a questi requisiti, ma in questo libro saranno discusse solo le due più importanti.

Logica domino

Concetti fondamentali Un modulo in logica domino consiste in un blocco dinamico di tipo *n* seguito in cascata da un invertitore CMOS statico (Figura 6-64) [Krambeck82]. Durante la fase di precarica l'uscita del blocco dinamico (ingresso dell'invertitore) è caricata fino a V_{DD} e l'uscita dell'invertitore assume il valore 0. Durante la fase di valutazione, se la configurazione degli ingressi è tale che il *pull-down* della porta dinamica sia acceso, il nodo di uscita viene scaricato a 0 e l'invertitore commuta da 0 a 1. Supponendo che tutte le uscite dei blocchi domino siano ingressi di un altro blocco domino,[6]

[5] Questa regola trascura l'effetto della ridistribuzione di carica e delle correnti di perdita precedentemente discusse.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

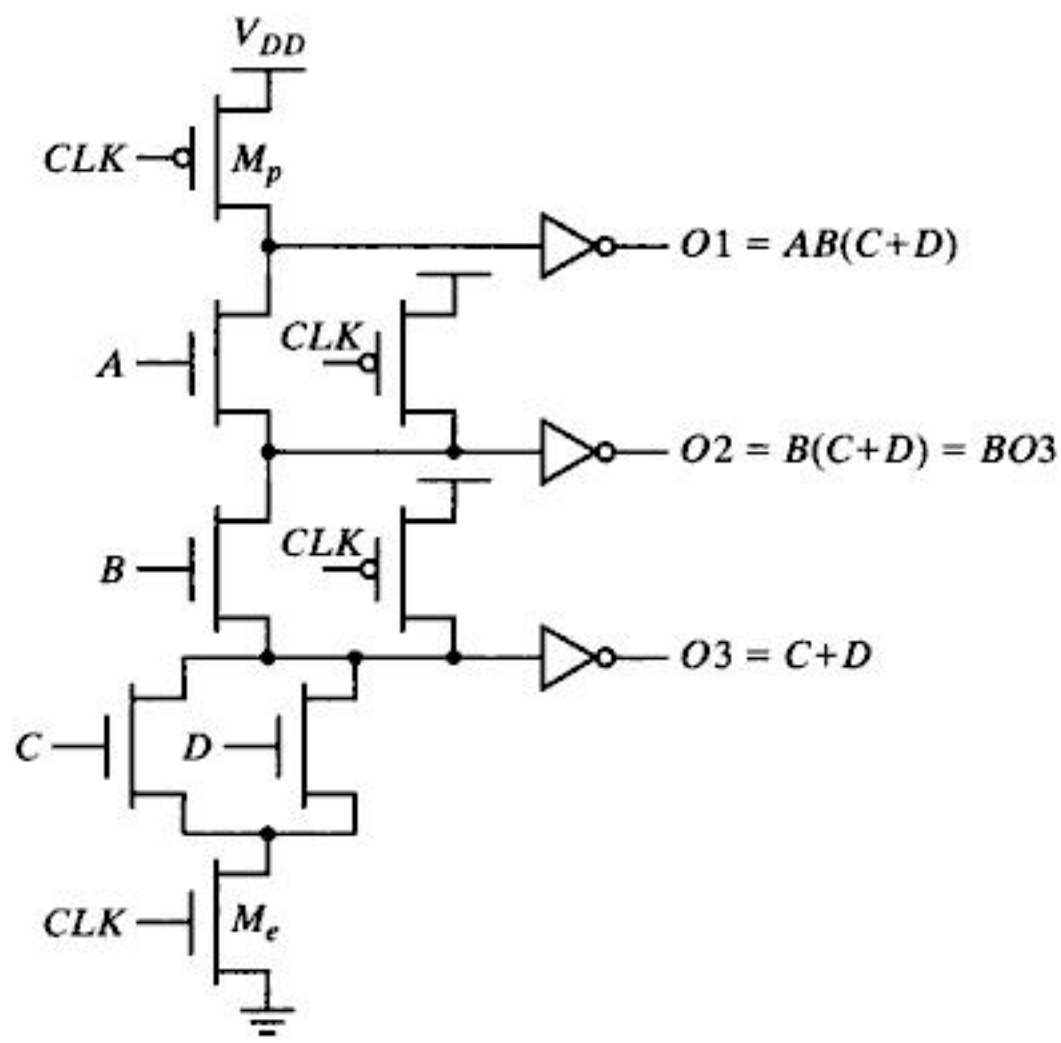


Figura 6-68 Logica domino a uscite multiple.

entrano in fase di precarica nello stesso istante e l'operazione di precarica impiega un tempo pari al tempo di propagazione di due porte logiche (caricamento dell'uscita della porta dinamica e tempo di propagazione alto-basso del segnale attraverso l'invertitore CMOS statico). Il cammino critico del circuito in fase di valutazione passa attraverso le reti di *pull-down* delle porte dinamiche e i transistor PMOS di *pull-up* degli invertitori. Quindi, per accelerare il funzionamento del circuito in valutazione il rapporto tra la larghezza dei transistor PMOS e NMOS dell'invertitore deve essere abbastanza alta in modo tale che la tensione di soglia logica sia vicina a V_{DD}. A tale scopo è necessario realizzare il transistor NMOS molto piccolo (dimensioni minime) e il transistor PMOS abbastanza largo. La dimensione del dispositivo NMOS influenza solo il tempo di precarica, che è in genere abbastanza contenuto, dato che l'operazione avviene in parallelo su tutte le porte. Il solo svantaggio di tale dimensionamento è la riduzione dei margini di rumore: quindi, nel progettare il circuito è necessario considerare il giusto compromesso tra immunità ai disturbi e prestazioni del circuito.

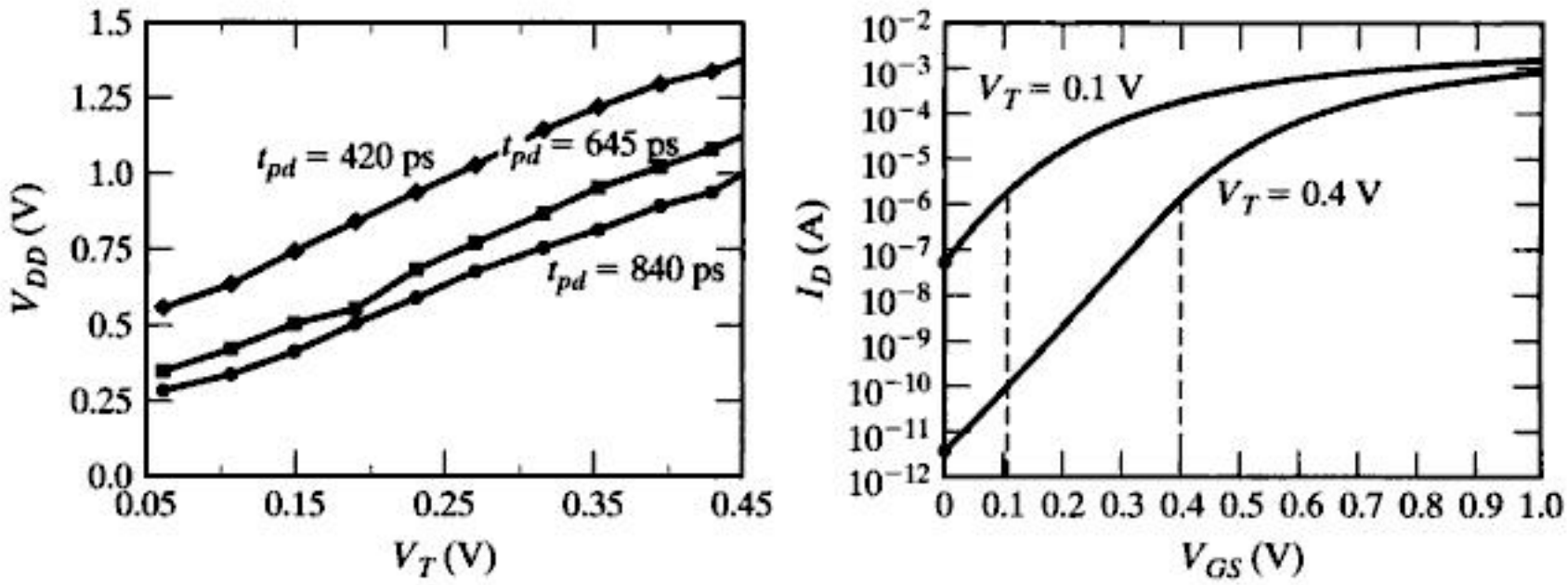
Sono state proposte diverse varianti della logica domino [Bernstein98]. Tra queste la *logica domino a uscite multiple* è una tecnica per ridurre l'occupazione di area. Il concetto alla base di questo approccio è schematizzato nella Figura 6-68 ed è in grado di realizzare funzioni logiche a singolo stadio con uscite multiple sfruttando il fatto che alcune di queste uscite sono funzione di un sottoinsieme delle altre uscite della funzione. In questo esempio, l'uscita $O3 = C + D$ è riutilizzata in tutte tre le uscite ed è quindi implementata nella parte più bassa della rete di *pull-down*: per esempio, $O2 = B \cdot (C + D) = B \cdot O3$ e per il calcolo può essere riutilizzata la sottorete logica che realizza la funzione $O3$. Notiamo che, per ottenere il risultato corretto, tutti i nodi interni che implementano una funzione logica devono essere precaricati a V_{DD}. In tal modo il numero di dispositivi di precarica non viene ridotto, essendo comunque uguale al

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) *rapporto V_{DD}/V_T per un dato tempo di ritardo* (b) *Corrente di perdita in funzione di V_T*

Figura 6-71 Effetto della tensione di alimentazione e della tensione di soglia sulle prestazioni e sulla corrente di perdita di un circuito.

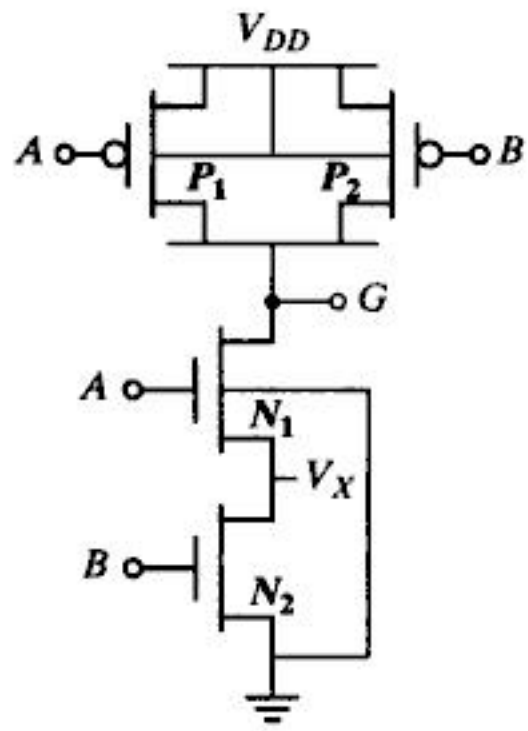
Questo compromesso non è esente da penalità. Riducendo la tensione di soglia, aumenta esponenzialmente la corrente di sottosoglia, come descritto dall'Eq. (3.39) (qui ripetuta per comodità):

$$I_{DSS} = I_S 10^{\frac{V_{GS}-V_{th}}{S}} \left(1 - 10^{-\frac{nV_{DS}}{S}}\right) \tag{6.46}$$

Nell'Eq. (6.46), S è la *pendenza di sottosoglia* del dispositivo. La corrente di sottosoglia di un invertitore è la corrente di perdita del transistor NMOS per $V_{in} = 0$ V e $V_{out} = V_{DD}$ (o quella del PMOS con $V_{in} = V_{DD}$ e $V_{out} = 0$). L'aumento esponenziale della corrente di perdita dell'invertitore al diminuire della tensione di soglia è illustrato nella Figura 6-71b.

Queste correnti di perdita costituiscono un serio problema sopratutto per i circuiti caratterizzati da brevi periodi di elaborazione alternati a lunghi periodi di inattività. Il processore di un telefono cellulare, per esempio, rimane in stato di attesa per la maggior parte del tempo. Quando il processore è in modalità di attesa, idealmente il consumo statico di potenza del sistema dovrebbe essere nullo o quasi. Questa ipotesi è vera solo se le correnti di perdita sono molto basse, cioè i dispositivi hanno una tensione di soglia molto elevata. Ciò è in netto contrasto con il panorama evolutivo che abbiamo appena descritto, dove ottenere prestazioni elevate con basse tensioni di alimentazione richiede la riduzione della tensione di soglia. Per soddisfare contemporanemaente le esigenze di elevate prestazioni durante i periodi di attività e di basso consumo statico durante i periodi di attesa, sono state introdotte diverse variazioni al processo di fabbricazione e tecniche per il controllo delle correnti di perdita. Molti processi di fabbricazione CMOS con risoluzione inferiore a 0.18 μm, consentono di realizzare dispositivi con diverse tensioni di soglia. Tipicamente è offerta la scelta tra due categorie: dispositivi con tensione di soglia bassa per i circuiti ad alte prestazioni; dispositivi con tensione di soglia più alta per i circuiti a basso consumo. Una tecnica che sta ottenendo un grande successo è il controllo dinamico della

Materiale protetto da copyright



A	B	V_X	I_{DSS}
0	0	$V_{th}\ln(1+n)$	$I^N_{DSS}(V_{GS}=V_{BS}=-V_X)$
0	1	0	$I^N_{DSS}(V_{GS}=V_{BS}=0)$
1	0	$V_{dd}-V_T$	$I^N_{DSS}(V_{GS}=V_{BS}=0)$
1	1	0	$2\,I^P_{DSS}(V_{SG}=V_{SB}=0)$

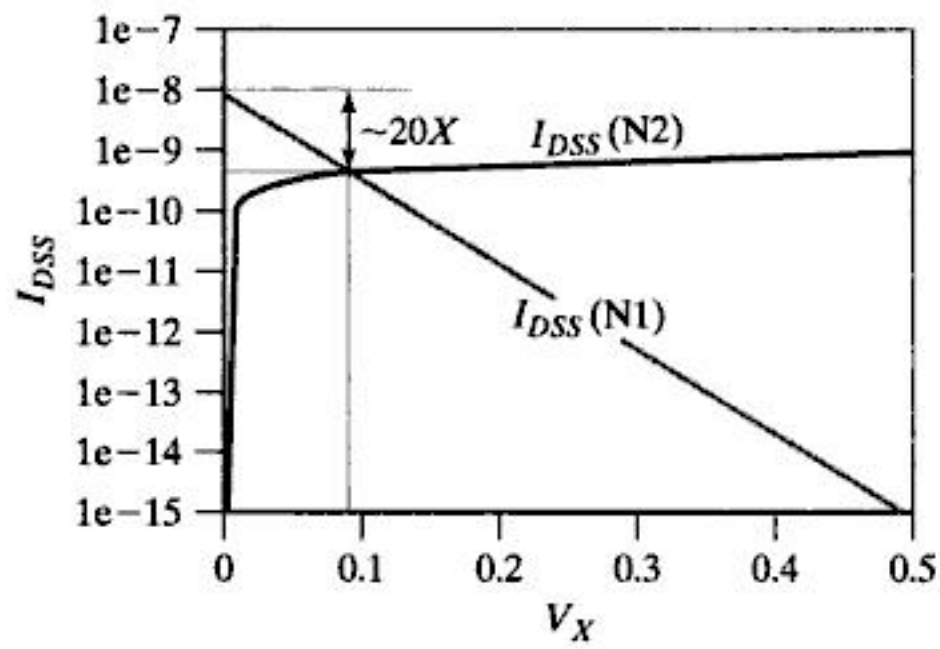


Figura 6-72 Riduzione della corrente di sottosoglia in una porta NAND a due ingressi (a) grazie alla serie di transistor per diverse configurazioni degli ingressi (b). La Figura (c) mostra la simulazione delle curve di carico della porta con $A = B = 0$.

tensione di soglia di un dispositivo sfruttando l'effetto body del transistor. L'uso di questa tecnica richiede, però, un processo di fabbricazione a doppia *well* (vedi Figura 2-2).

Un oculato progetto può aiutare nel ridurre tutte quelle correnti di perdita che sono funzione della topologia circuitale e del livello di tensione degli ingressi. Poiché la tensione di soglia V_T dipende dal potenziale del substrato (V_{BS}), la corrente di sottosoglia di un transistor MOS dipende non solo dalla tensione V_{GS}, ma anche dal potenziale di substrato. In un invertitore con $In = 0$, la corrente di sottosoglia è determinata dal transistor NMOS con $V_{GS} = V_{BS} = 0$ V. In porte logiche più complesse, come la porta NAND a due ingressi della Figura 6-72, la corrente di perdita dipende dalla combinazione degli ingressi. La corrente di sottosoglia è minima quando $A = B = 0$. In queste condizioni, il nodo intermedio X assume la tensione

$$V_X \approx V_{th}\ln(1+n) \tag{6.47}$$

La corrente di perdita della porta è determinata quindi dal transistor NMOS più alto con $V_{GS} = V_{BS} = -V_X$. Chiaramente la corrente di sottosoglia in queste condizioni è inferiore a quella del semplice invertitore. La riduzione di corrente dovuta ai due transistor connessi in

Materiale protetto da copyright

serie è detta *effetto serie (stack effect)*. La tabella mostrata nella Figura 6-72 elenca le varie componenti della corrente di perdita nella porta NAND a due ingressi per tutte le combinazioni degli stessi.

La realtà è ancora migliore: in transistor MOS a canale corto, la corrente di sottosoglia dipende non solo dalla tensione gate (V_{GS}) e dal potenziale del substrato (V_{BS}), ma anche dalla tensione di drain (V_{DS}). La tensione di soglia di un MOS a canale corto diminuisce all'aumentare di V_{DS} a causa della riduzione della barriera di potenziale tra source e drain (fenomeno noto come DIBL (*drain-induced barrier lowering*). Tipicamente il DIBL può indurre variazioni in V_T da 20 a 150 mV in seguito alla variazione di V_{DS}. A causa di ciò, l'impatto dell'effetto serie è ancor può significativo in transistor a canale corto. Il potenziale del nodo intermedio riduce la tensione drain-source del transistor più alto, aumentandone la tensione di soglia e riducendo la corrente di sottosoglia.

Esempio 6.19 Effetto serie in una porta NAND a due ingressi

Consideriamo ancora la porta NAND a due ingressi della Figura 6-72a, quando sia N_1 sia N_2 sono interdetti ($A = B = 0$). Dalla curva di carico simulata mostrata in Figura 6-72c, vediamo che V_X assume un valore approssimativamente di 100 mV in condizioni stazionarie. La corrente di sottosoglia in condizioni stazionarie è determinata dalle tensioni $V_{GS} = V_{BS} = -100$ mV e $V_{DS} = V_{DD} - 100$ mV, che è 20 volte più piccola che la corrente di sottosoglia di un transistor NMOS da solo con $V_{GS} = V_{BS} = 0$ mV e $V_{DS} = V_{DD}$ [Ye98].

In sintesi, la corrente di sottosoglia in circuiti complessi è significativamente minore che in un singolo dispositivo preso separatamente. Osserviamo che la massima riduzione si verifica quando tutti i transistor nella serie sono spenti e il nodo intermedio raggiunge il valore stazionario. Sfruttare questo effetto richiede un'attenta selezione della posizione dei segnali durante i periodi di attesa e inattività del circuito.

Problema 6.9 Calcolo di V_X

L' Eq. (6.47) calcola il potenziale del nodo intermedio per una porta NAND a due ingressi a meno di un errore del 10%, per $A = B = 0$. Derivare l'Eq. (6.47) assumendo che (1) V_T e I_S di N_1 e N_2 siano approssimativamente uguali, (2) i transistor NMOS abbiano le stesse dimensioni e (3) $n < 1.5$.

6.5 Riepilogo

In questo capitolo abbiamo studiato approfonditamente il comportamento e le prestazioni dei circuiti digitali combinatori in logica CMOS in termini di area, velocità e consumo di potenza. Riassumiamo di seguito i punti salienti:

- La *logica CMOS statica complementare* combina reti di *pull-down* e *pull-up* duali, una e solo una delle quali è abilitata in ogni istante.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

ad alte prestazioni, [Shoji96] e [Bernstein98] offrono più approfondita discussione sull'ottimizzazione e l'analisi dei circuiti digitali MOS. Il problema del consumo di potenza è relativamente nuovo, ma sono ugualmente disponibili dei riferimenti in [Chandrakasan95], [Rabaey95], e [Pedram02].

Lavori innovativi sul campo della logica MOS sono tipicamente pubblicati negli atti di conferenze quali *ISSCC* e *VLSI circuits symposium* oppure in riviste come *IEEE Journal of Solid State Circuits* (specialmente nel numero di novembre).

Riferimenti scientifici

[Bernstein98] K. Bernstein et al., *High-Speed CMOS Design Styles*, Kluwer Academic Publishers, 1998.

[Chandrakasan95] A. Chandrakasan and R. Brodersen, *Low Power Digital CMOS Design*, Kluwer Academic Publishers, 1995.

[Chandrakasan01] A. Chandrakasan, W. Bowhill, and F. Fox, ed., *Design of High-Performance Microprocessor Circuits*, IEEE Press, 2001.

[Gonçalvez83] N. Gonçalvez and H. De Man, "NORA: A Racefree Dynamic CMOS Technique for Pipelined Logic Structures," *IEEE Journal of Solid State Circuits*, vol. SC-18, no. 3, pp. 261–266, June 1983.

[Heller84] L. Heller et al., "Cascade Voltage Switch Logic: A Differential CMOS Logic Family," *Proc. IEEE ISSCC Conference*, pp. 16–17, February 1984.

[Krambeck82] R. Krambeck et al., "High-Speed Compact Circuits with CMOS," *IEEE Journal of Solid State Circuits*, vol. SC-17, no. 3, pp. 614–619, June 1982.

[Landman91] P. Landman and J. Rabaey, "Design for Low Power with Applications to Speech Coding," *Proc. International Micro-Electronics Conference*, Cairo, December 1991.

[Parameswar96] A. Parameswar, H. Hara, and T. Sakurai, "A Swing Restored Pass-Transistor Logic-Based Multiply and Accumulate Circuit for Multimedia Applications," *IEEE Journal of Solid State Circuits*, vol. SC-31, no. 6, pp. 805–809, June 1996.

[Pedram02] M. Pedram and J. Rabaey, ed., *Power-Aware Design Methodologies*, Kluwer, 2002.

[Rabaey95] J. Rabaey and M. Pedram, ed., *Low Power Design Methodologies*, Kluwer, 1995.

[Radhakrishnan85] D. Radhakrishnan, S. Whittaker, and G. Maki, "Formal Design Procedures for Pass-Transistor Switching Circuits," *IEEE Journal of Solid State Circuits*, vol. SC-20, no. 2, pp. 531–536, April 1985.

[Shoji88] M. Shoji, *CMOS Digital Circuit Technology*, Prentice Hall, 1988.

[Shoji96] M. Shoji, *High-Speed Digital Circuits*, Addison-Wesley, 1996.

[Sutherland99] I. Sutherland, B. Sproull, and D. Harris, *Logical Effort*, Morgan Kaufmann, 1999.

[Weste93] N. Weste and K. Eshragian, *Principles of CMOS VLSI Design: A Systems Perspective*, Addison-Wesley, 1993.

[Yano90] K. Yano et al., "A 3.8 ns CMOS 16 × 16 b Multiplier Using Complimentary Pass-Transistor Logic," *IEEE Journal of Solid State Circuits*, vol. SC-25, no. 2, pp. 388–395, April 1990.

[Ye98] Y. Ye, S. Borkar, and V. De, "A New Technique for Standby Leakage Reduction in High-Performance Circuits," *Symposium on VLSI Circuits*, pp. 40–41, 1998.

Esercizi

Per una raccolta aggiornata di problemi sulla logica CMOS consultare la pagina **http://bwrc.eecs.berkeley.edu/IcBook**.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

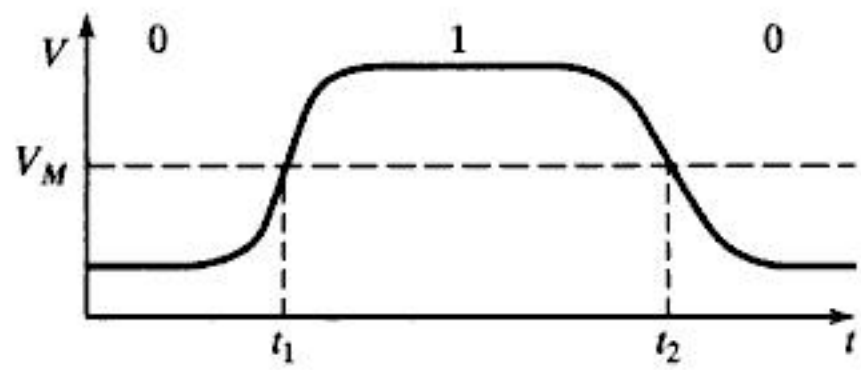


Figura C-1 Trasformazione della variabile temporale in forma discreta.

Questa rappresentazione ternaria {0, 1, *X*} è largamente usata nei simulatori sia a livello di porta logica, sia a livello di transistor. Espandendo l'insieme dei valori permessi per i dati, possiamo ottenere informazioni più dettagliate, pur conservando la capacità di maneggiare sistemi complessi. Alcune possibili estensioni sono il valore *Z* per indicare che un nodo è in alta impedenza, *R* e *F* per indicare che il nodo sta eseguendo una salita e una discesa rispettivamente. Alcuni simulatori commerciali offrono fino a dodici stati possibili.

Sebbene rendere i segnali discreti porti a un sostanziale miglioramento nelle prestazioni del simulatore, benefici molto simili si possono ottenere rendendo discreta la variabile temporale. Consideriamo la forma d'onda nella Figura C-1, che rappresenta il segnale all'ingresso di un invertitore con una tensione di soglia logica V_M. È ragionevole ipotizzare che l'uscita dell'invertitore commuterà con un certo tempo di ritardo solo dopo che l'ingresso attraversa il valore V_M. Non essendo strettamente interessati alla forma d'onda del segnale, è sufficiente valutare il circuito solamente nei due istanti temporali di interesse t_1 e t_2.

Analogamente, i punti di interesse relativi al segnale di uscita sono localizzati in $t_1 + t_{pHL}$ e $t_2 + t_{pLH}$. Un simulatore che analizza e valuta il circuito solo nell'istante in cui avviene una commutazione dei suoi ingressi è detto *simulatore ad eventi discreti (event-driven simulator)*. L'ordine con cui sono considerati gli eventi è determinato inserendo tutti gli eventi programmati in una coda temporale e poi elaborandoli in ordine cronologico. Supponiamo che la forma d'onda della Figura C-1 sia il segnale di ingresso a una data porta logica. Il primo evento è previsto all'istante t_1. Dopo che questa commutazione è stata analizzata, un nuovo evento è previsto al suo nodo di uscita all'istante $t_1 + t_{pHL}$ ed è inserito nella coda degli eventi. Questo approccio basato sugli eventi è evidentemente più efficiente della tecnica usata nei simulatori circuitali, in cui le forme d'onda vengono campionate a intervalli di tempo fissi e costanti. Per tenere traccia dell'effetto del fan-out, il tempo di propagazione di un circuito può essere facilmente espresso in termini di tempo di propagazione intrinseco (t_{in}) e di contributo dipendente dal carico (t_l), e può essere differenziato in base al tipo di transizione (basso-alto o alto basso):

$$t_{pLH} = t_{inLH} + t_{lLH} \times C_L \tag{C.1}$$

La capacità C_L può essere espressa sia in valore assoluto (per esempio, in picofarad) sia come fan-out, cioè il numero di porte equivalenti (uguali alla porta in questione). Osserviamo che questo metodo assomiglia molto al modello del *lavoro logico* introdotto nel Capitolo 6.

Materiale protetto da copyright

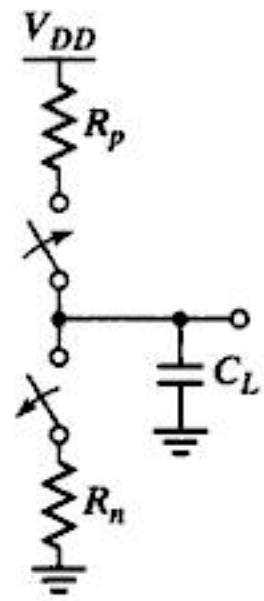


Figura C-2 Modello a interruttore di un invertitore CMOS.

Sebbene questa tecnica offra molti benefici in termini di prestazioni, rimane lo svantaggio che gli eventi possono accadere ad ogni istante. Un'altra semplificazione potrebbe essere quella di rendere la variabile temporale non solo discreta, ma anche quantizzata, permettendole di assumere solo valori che siano multipli interi di una certa *unità di tempo*. Un esempio di questo approccio è il modello a *ritardo unitario*, dove ciascun circuito possiede un tempo di ritardo fissato uguale ad *una unità*. Per finire, il modello più semplice è quello a *ritardo zero*, dove tutte le porte logiche si assumono prive di ritardo. In queste condizioni, il tempo procede da un ciclo di clock al successivo e tutti gli eventi nel sistema accadono istantaneamente, a partire dall'arrivo del segnale di clock. Questi concetti possono essere applicati ai successivi livelli di astrazione dando luogo alle tecniche di simulazione discusse nel seguito.

Simulazione a livello di interruttore (switch-level)

La natura non lineare dei dispositivi a semiconduttore è uno dei maggiori ostacoli alla velocità della simulazione. Il modello a interruttore (*switch-level model*) [Bryant81] scavalca l'ostacolo approssimando il comportamento del transistor con quello di una resistenza lineare il cui valore è funzione delle condizioni operative. Se il transistor è spento, la resistenza ha valore infinito; se il transistor è acceso, si assume un valore medio della resistenza del dispositivo (Figura C-2). Il circuito diventa quindi una rete lineare di resistenze e capacità a parametri tempo-varianti, che permette un'analisi semplice ed efficiente. La risoluzione della rete di resistori fornisce come risultato il valore logico stazionario del segnale impiegando tipicamente il modello $\{0, 1, X\}$. Se, per esempio, se la resistenza equivalente tra il nodo di uscita e la massa è sostanzialmente inferiore a quella tra l'uscita e V_{DD}, il valore logico del nodo è posto a 0. La temporizzazione degli eventi può essere risolta analizzando la rete RC. In alternativa, è possibile usare il semplice modello del ritardo unitario.

Esempio C.1 Confronto tra simulazione a livello di circuito e di interruttore

Un sommatore a quattro bit viene simulato a livello di interruttore impiegando il simulatore IRSIM ([Salz89]). I risultati sono riportati nella Figura C-3a. Inizialmente, tutti

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

```
entity add is
    port (
        IN0 : in bit_vector(15 downto 0);
        IN1 : in bit_vector(15 downto 0);
        OUT0 : out bit_vector(15 downto 0)
    );
end add;

architecture behavior of add is
begin
    process(IN0, IN1)
        variable C : bit_vector(16 downto 0);
        variable S : bit_vector(15 downto 0);
    begin
        loop1:
        for i in 0 to 15 loop
            S(i) := IN0(i) xor IN1(i) xor C(i);
            C(i+1):= IN0(i) and IN1(i) or C(i) and (IN0(i) or IN1(i));
        end loop loop1;
        OUT0 <= S;
    end process;
end behavior;
```

Figura C-5 Descrizione comportamentale di un sommatore a 16 bit.

entità è possibile tenere conto di un valore complessivo del tempo di ritardo, sebbene il tempo di ritardo effettivo possa variare da bit a bit.

Comunemente si distinguono due tipi di descrizione di un elemento: descrizione *funzionale* (o *strutturale*) e descrizione *comportamentale*. La descrizione a livello funzionale rispecchia la struttura circuitale. La descrizione a livello comportamentale ne descrive solo la relazione ingresso-uscita. Il tempo di ritardo perde di significato e la simulazione viene normalmente eseguita usando come costante di tempo il ciclo di clock o multipli di essa. Per esempio, usando il modello comportamentale di un microprocessore, per verificare la correttezza delle sue istruzioni è possibile usare come unità temporale il periodo di esecuzione di un'istruzione.

I linguaggi più comunemente usati a questo livello di astrazione sono VHDL e VERILOG. VHDL permette l'introduzione di tipi di dati definiti dall'utente come vettori a 16 bit in complemento a due ed enumerazioni. Per implementare i modelli comportamentali al primo ordine, molti progettisti tendono ad usare linguaggi di programmazione tradizionali come il Linguaggio C o C++. Questo approccio ha il vantaggio di offrire maggior flessibilità, ma richiede all'utente di definire tutti i tipi di dati e di scrivere lo scenario completo della simulazione.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Riferimenti scientifici

[Agrawal90] P. Agrawal and W. Dally, "A Hardware Logic Simulation System," *IEEE Trans. Computer-Aided Design*, CAD-9, no. 9, pp. 19–29, January 1990.

[Bryant81] R. Bryant, *A Switch-Level Simulation Model for Integrated Logic Circuits*, Ph. D. diss., MIT Laboratory for Computer Science, report MIT/LCS/TR-259, March 1981.

[Salz89] A. Salz and M. Horowitz, "IRSIM: An Incremental MOS Switch-Level Simulator," *Proceedings of the 26th Design Automation Conference*, pp. 173–178, 1989.

[SystemC] *Everything You Wanted to Know about SystemC, http://www.systemc.org/*

[Thor88] R. Alverson et al., "THOR User's Manual," Technical Report CSL-TR-88-348 and 349, Stanford University, January 1988.

[TimeMill] *http://www.synopsys.com/products/mixedsignal/mixedsignal.html*

[VHDL88] VHDL Standards Committee, IEEE Standard VHDL Language Reference Manual, IEEE standard 1076–1077, 1978.

[White87] J. White and A. Sangiovanni-Vincentelli, *Relaxation Techniques for the Simulation of VLSI Circuits*, Kluwer Academic, 1987.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

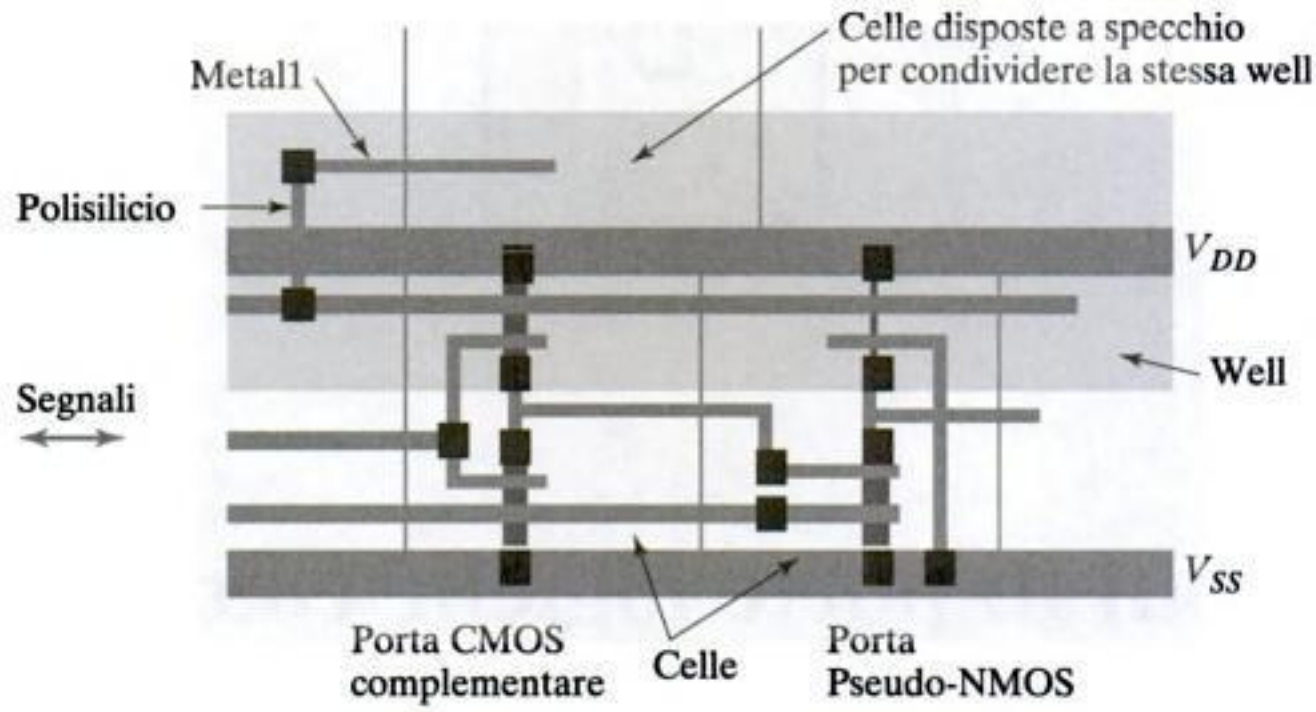


Figura D-1 Tecnica Weinberger per il layout di porte logiche complesse (usando un singolo strato di interconnessioni metalliche).

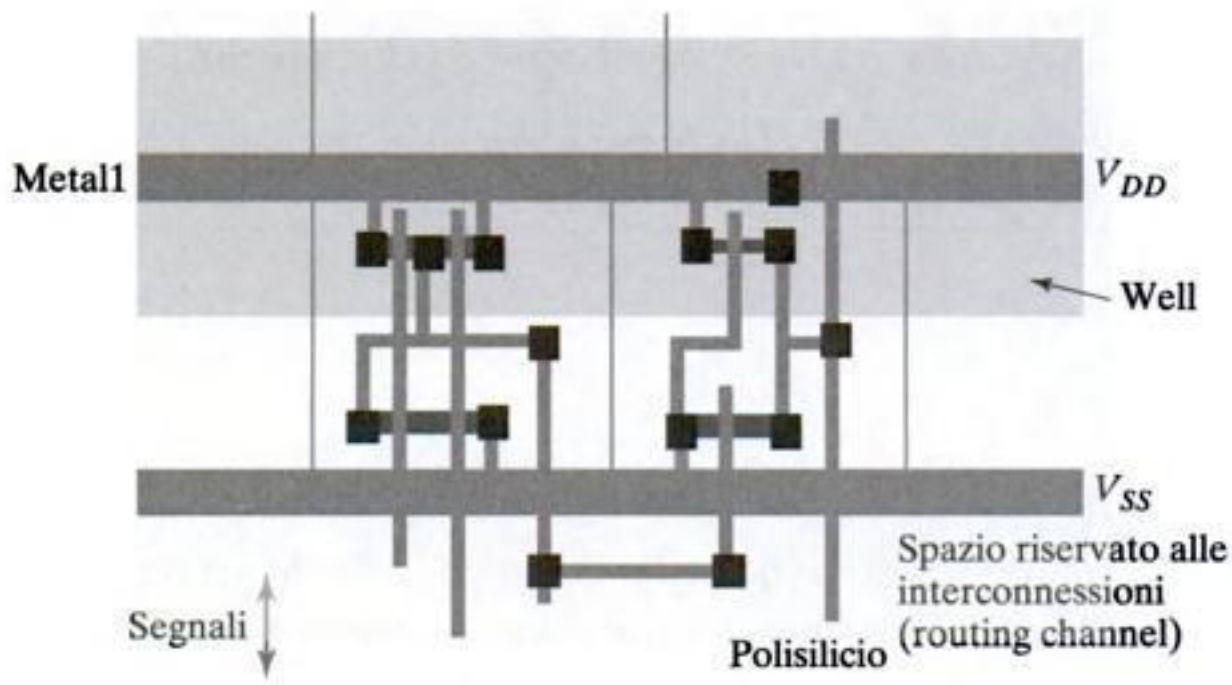


Figura D-2 Layout di una porta logica mediante celle standard.

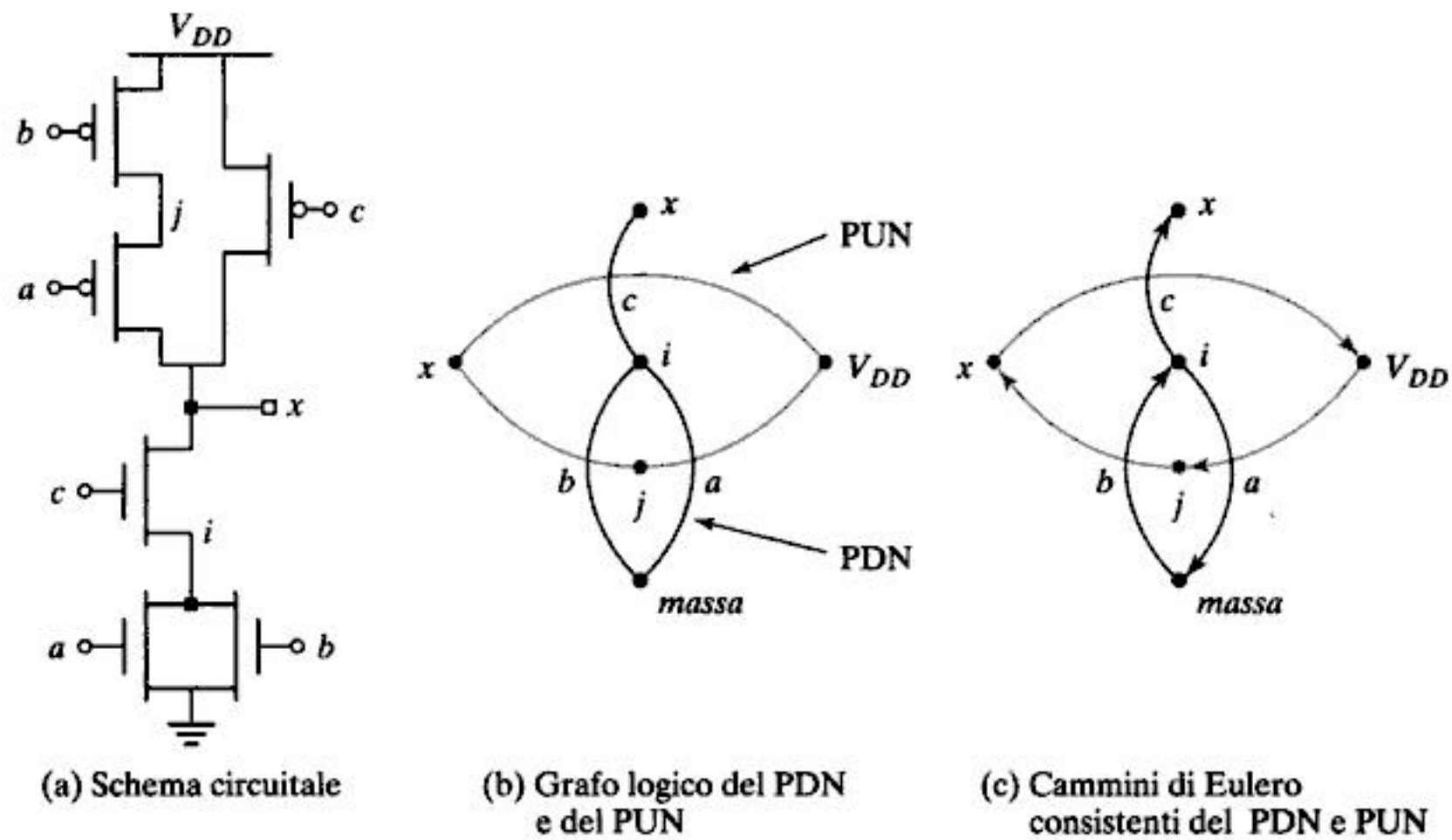
Nella tecnica a *celle standard*, i segnali sono portati lungo il circuito mediante piste di polisilicio perpendicolari alle linee di distribuzione della massa e dell'alimentazione (Figura D-2). Questo approccio fornisce un layout più denso e compatto per le porte logiche CMOS statiche, poiché le piste verticali di polisilicio fungono da ingressi per entrambi i transistor NMOS e PMOS. Un esempio di circuito realizzato con la tecnica a celle standard è mostrato nella Figura 6-12. L'interconnessione tra celle è generalmente realizzata in regioni riservate a tale scopo (*routing channel*), come illustrato nella Figura D-2. La tecnica a *celle standard* è attualmente molto popolare grazie al suo elevato grado di automazione (per una descrizione dettagliata dei vari strumenti di progettazione automatica che supportano la tecnica a *celle standard* rimandiamo al Capitolo 8).

Metodo dei cammini di Eulero

Vale la pena di analizzare come le funzioni logiche complesse possano essere realizzate efficacemente mediante questa strategia di layout. Per aumentare la densità di integrazione è

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Schema circuitale (b) Grafo logico del PDN e del PUN (c) Cammini di Eulero consistenti del PDN e PUN

Figura D-4 Schema circuitale, grafo logico e cammini di Eulero della funzione booleana $x = (a + b) \cdot c$.

PUN della funzione booleana $\overline{x} = (a + b) \cdot c$ (notiamo che questa tecnica può essere usata per derivare la rete duale).

2. **Identificazione dei cammini di Eulero.** Un cammino di eulero in un grafo è definito come un cammino che attraversa tutti i nodi del grafo percorrendo ciascun arco una sola volta. L'identificazione di questi cammini è importante, perché un riordino degli ingressi che conduce a una striscia ininterrotta di regione attiva di tipo *n* (o di tipo *p*) è possibile solo se esiste un cammino di Eulero nel grafo logico della rete di PDN (o PUN). Il ragionamento che sta dietro a questo risultato è descritto di seguito.

Per creare una striscia ininterrotta di regione attiva, tutti i transistor devono essere visitati in sequenza; cioè, il drain di un dispositivo coincide con il source di quello successivo. Ciò equivale ad attraversare il grafo logico lungo un cammino di Eulero. Ma i cammini di Eulero non sono univoci: potrebbero esistere molte soluzioni differenti.

La sequenza di archi in un cammino di Eulero coincide con l'ordine degli ingressi nel layout della porta logica. Per ottenere lo stesso ordine nelle reti di PDN e PUN, requisito necessario se vogliamo usare una sola pista di polisilicio per ciascuno dei segnali di ingresso, i due cammini di Eulero devono essere *coerenti*, cioè, devono avere la stessa sequenza.

La coppia di cammini di Eulero coerenti per l'esempio della Figura D-4a è mostrata nella Figura D-4c e la Figura D-3b mostra il layout corrispondente. Uno sguardo al grafo della funzione mostra che $\{a\ c\ b\}$ è un cammino di Eulero per il PUN, ma non per il PDN e, quindi, non ammette una soluzione con una striscia ininterrotta di regione attiva (Figura D-3a).

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

orizzontali multiple di regione attiva allineate verticalmente una sopra l'altra. In tal caso una singola pista di polisilicio può essere usata come linea di ingresso comune a tutti i transistor. Ciò potrebbe essere particolarmente indicato per alcune strutture, come il layout di una porta logica XNOR. È quindi raccomandabile un'analisi caso per caso.

Approfondimenti

Una buona panoramica sulle tecniche di generazione può essere trovata in [Rubin87, pp. 116–128]. Alcuni dei lavori che sono un punto di riferimento in questo campo sono elencati di seguito:

[Clein00] D. Clein, *CMOS IC Layout*, Newnes, 2000.

[Rubin87] S. Rubin, *Computer Aids for VLSI Design*, Addison-Wesley, 1987.

[Uehara81] T. Uehara and W. Van Cleemput, "Optimal Layout of CMOS Functional Arrays," *IEEE Trans. on Computers*, vol. C-30, no. 5, pp. 305–311, May 1981.

[Weinberger67] A. Weinberger, "Large Scale Integration of MOS Complex Logic: A Layout Method," *IEEE Journal of Solid State Circuits*, vol. 2, no. 4, pp. 182–190, 1967.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

7.6.1 Il trigger di Schmitt
7.6.2 I circuiti monostabili
7.6.3 I circuiti astabili
7.7 Le prospettive: scelta della struttura di temporizzazione
7.8 Riepilogo
7.9 Approfondimenti

7.1 Introduzione

Come abbiamo visto in precedenza, i circuiti logici combinatori sono caratterizzati dal fatto che il valore dell'uscita di un blocco logico è unicamente funzione dei valori in ingresso all'istante *presente*, se si assume che sia trascorso tempo a sufficienza affinché le porte logiche siano a regime. In realtà, la quasi totalità dei sistemi realmente utilizzabili necessita della memorizzazione di informazione, caratteristica di un'altra classe di circuiti: i *circuiti logici sequenziali*. In questi circuiti, l'uscita non dipende solo dal valore *presente* degli ingressi, ma anche da valori *precedenti*. In altre parole, un circuito sequenziale ricorda una parte delle storia passata del sistema, cioè ha memoria.

La Figura 7-1 mostra lo schema a blocchi di una generica *macchina a stati finiti* che è costituita da logica combinatoria e registri, i quali sono in grado di memorizzare lo stato del sistema. Il sistema mostrato in questa figura appartiene alla classe dei sistemi sequenziali *sincroni*, nei quali tutti i registri sono controllati da un segnale di temporizzazione globale (CLK). Le uscite della macchina a stati finiti sono funzione degli *ingressi* presenti e dello *stato* presente. Il valore dello *stato futuro* è calcolato in base al valore *presente* dello *stato* e degli *ingressi* ed è quindi applicato agli ingressi dei registri. In corrispondenza del fronte di salita del clock, il valore dei bit che rappresentano lo *stato futuro* è memorizzato nei registri (dopo un certo tempo di propagazione) e quindi comincia un nuovo ciclo. In seguito i registri ignorano qualsiasi cambiamento nei loro segnali di ingresso finché non si verifica la commutazione del clock. I registri possono essere sensibili a l'uno o all'altro fronte del clock: convenzionalmente, i registri sensibili (cioè che campionano il dato) in corrispondenza al fronte di salita (positivo) del clock, sono detti *positive edge triggered*. Invece i registri sensibili al fronte di discesa (negativo) del clock sono detti *negative edge triggered*.

Questo capitolo tratta dell'implementazione in tecnologia CMOS dei blocchi sequenziali fondamentali più importanti. Sono possibili molteplici scelte sia rispetto ai blocchi sequenziali sia rispetto alle metodologie di sincronizzazione; operare la scelta più opportuna sta divenendo sempre più importante nei circuiti digitali moderni e può avere conseguenza significative in termini di prestazioni, potenza e complessità progettuale. Prima di affrontare nel dettaglio tutte queste opzioni, è necessario rivedere le figure di merito più significative e la classificazione degli elementi circuitali sequenziali.

7.1.1 Caratteristiche temporali dei circuiti sequenziali

Ci sono tre parametri importanti rispetto alle caratteristiche temporali di un registro, e sono mostrati nella Figura 7-2. Il *tempo di setup* (t_{su}) è il tempo per cui il dato di ingresso (D) deve

Materiale protetto da copyright

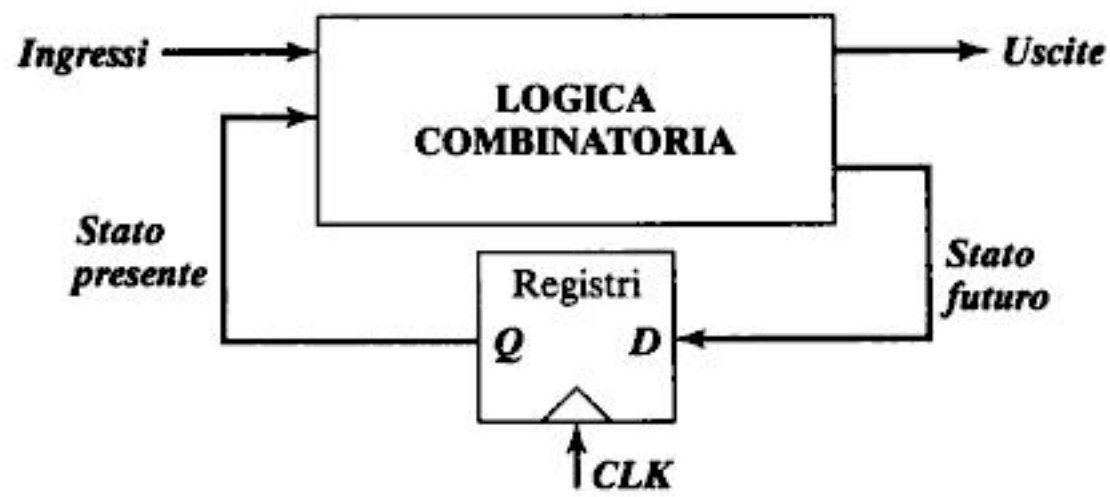


Figura 7-1 Schema a blocchi di una macchina a stati finiti che utilizza registri *positive edge-triggered*.

essere valido prima della transizione del clock (cioè, la transizione $0 \rightarrow 1$ per un registro *positive edge-triggered*). Il *tempo di hold* (t_{hold}) è il tempo per cui il dato di ingresso deve rimanere valido dopo la transizione del clock. Nell'ipotesi che i tempi di *setup* e *hold* siano rispettati, il dato all'ingresso D viene copiato all'uscita Q dopo un *tempo di propagazione* al caso peggiore (rispetto al fronte del clock) indicato con t_{c-q}.

Una volta noti i parametri temporali per i registri e per la logica combinatoria, è possibile ricavare i vincoli temporali a livello di sistema (vedi Figura 7-1 per una semplice visione a livello di sistema). Nei circuiti sequenziali sincroni, gli eventi di commutazione si verificano contemporaneamente in corrispondenza dello stimolo del clock e quindi i risultati delle operazioni logiche attendono la transizione del clock successiva prima di propagarsi alla stadio successivo. In altre parole, il ciclo successivo non può cominciare prima che tutte le operazioni presenti siano concluse e il sistema sia a regime. Il *periodo di clock* T a cui il circuito sequenziale funziona deve quindi comprendere il massimo ritardo di ogni stadio del sistema. Ipotizziamo che il tempo di propagazione massimo nella logica combinatoria sia t_{plogic}, mentre il ritardo minimo (chiamato anche *tempo di contaminazione*) sia t_{cd}. Il periodo di clock minimo T tale da non compromettere il funzionamento corretto del circuito è dato da

$$T \geq t_{c-q} + t_{plogic} + t_{su} \tag{7.1}$$

Il *tempo di hold* del registro impone un vincolo aggiuntivo per il corretto funzionamento, cioè

$$t_{cdregister} + t_{cdlogic} \geq t_{hold} \tag{7.2}$$

Dove $t_{cdregister}$ è il *tempo di propagazione* (o il *tempo di contaminazione*) minimo del registro. Questo vincolo assicura che il dato in ingresso all'elemento sequenziale venga mantenuto costante abbastanza a lungo dopo la commutazione del clock, senza essere modificato prematuramente da un nuovo flusso di dati in ingresso.

Come si vede dall'Eq. (7.1), è importante minimizzare i parametri temporali dei registri, dato che questi influiscono direttamente sulla frequenza massima a cui può operare il cir-

Materiale protetto da copyright

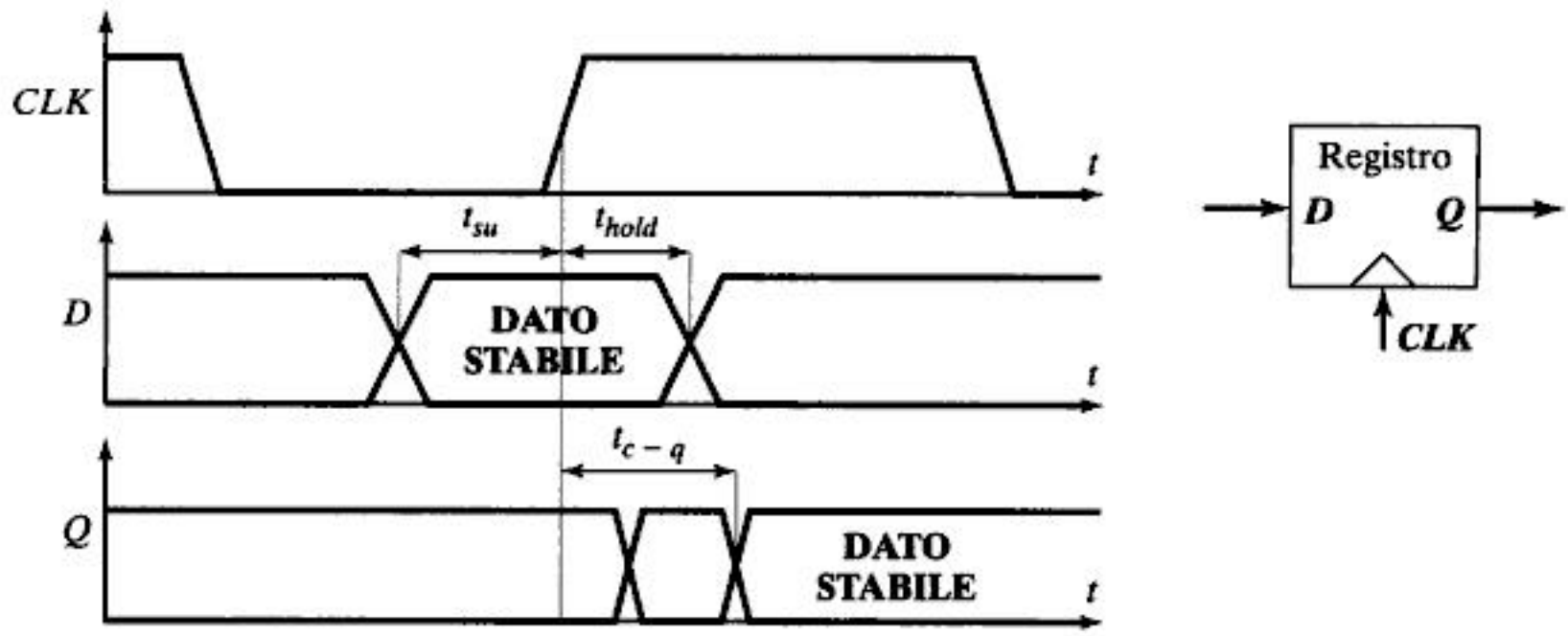


Figura 7-2 Definizione del *tempo di setup*, *tempo di hold* e *tempo di propagazione* di un registro sincrono.

cuito sequenziale. Di fatto, nei sistemi moderni ad alte prestazioni, si tende a mantenere molto ridotta la profondità della logica combinatoria e quindi i tempi di *propagazione* e *setup* dei registri costituiscono una frazione importante del periodo di clock minimo. Ad esempio, il microprocessore DEC Alpha EV6 [Gieseke97] ha una profondità massima di 12 porte logiche e il contributo dei registri al periodo minimo è circa il 15%. Generalmente, il vincolo dell'Eq. (7.2) non è difficile da rispettare, a meno che vi siano pochissime porte logiche, o non ve ne siano affatto, tra i registri.[1]

7.1.2 Classificazione degli elementi di memoria

Memoria locale o globale

A livello di sistema possiamo distinguere tra memoria globale e locale. La memoria integrata con i blocchi logici è detta *memoria locale* ed è generalmente costituita da singoli registri o al più da banchi di registri. Quando invece si fa riferimento a grossi nuclei di memoria centralizzata, si parla allora di *memoria globale*. Nelle memorie globali, trattate nel Capitolo 12, si raggiungono maggiori densità di integrazione, sfruttando opportunamente strutture a matrice e individuando il giusto compromesso tra dimensione, prestazioni e robustezza. In questo capitolo ci occuperemo delle memorie locali.

Memorie statiche o dinamiche

Le memorie possono essere di tipo statico oppure dinamico. Le memorie statiche conservano il loro stato fino a che rimangono alimentate. Sono costruite sfruttando la *retroazione positiva* e la rigenerazione, ottenuta usando topologie circuitali in cui vi sono delle connessioni intenzionali tra l'ingresso e l'uscita di porte combinatorie. Le memorie statiche sono sicuramente le più adatte quando lo stato del registro non deve essere aggiornato per lunghi periodi di tempo. Un buon esempio di questo tipo di dati statici è la configurazione di un sistema,

[1] O quando i segnali di clock applicati a registri diversi sono sfasati a causa dello skew. Tratteremo questo argomento specifico nel Capitolo 10.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

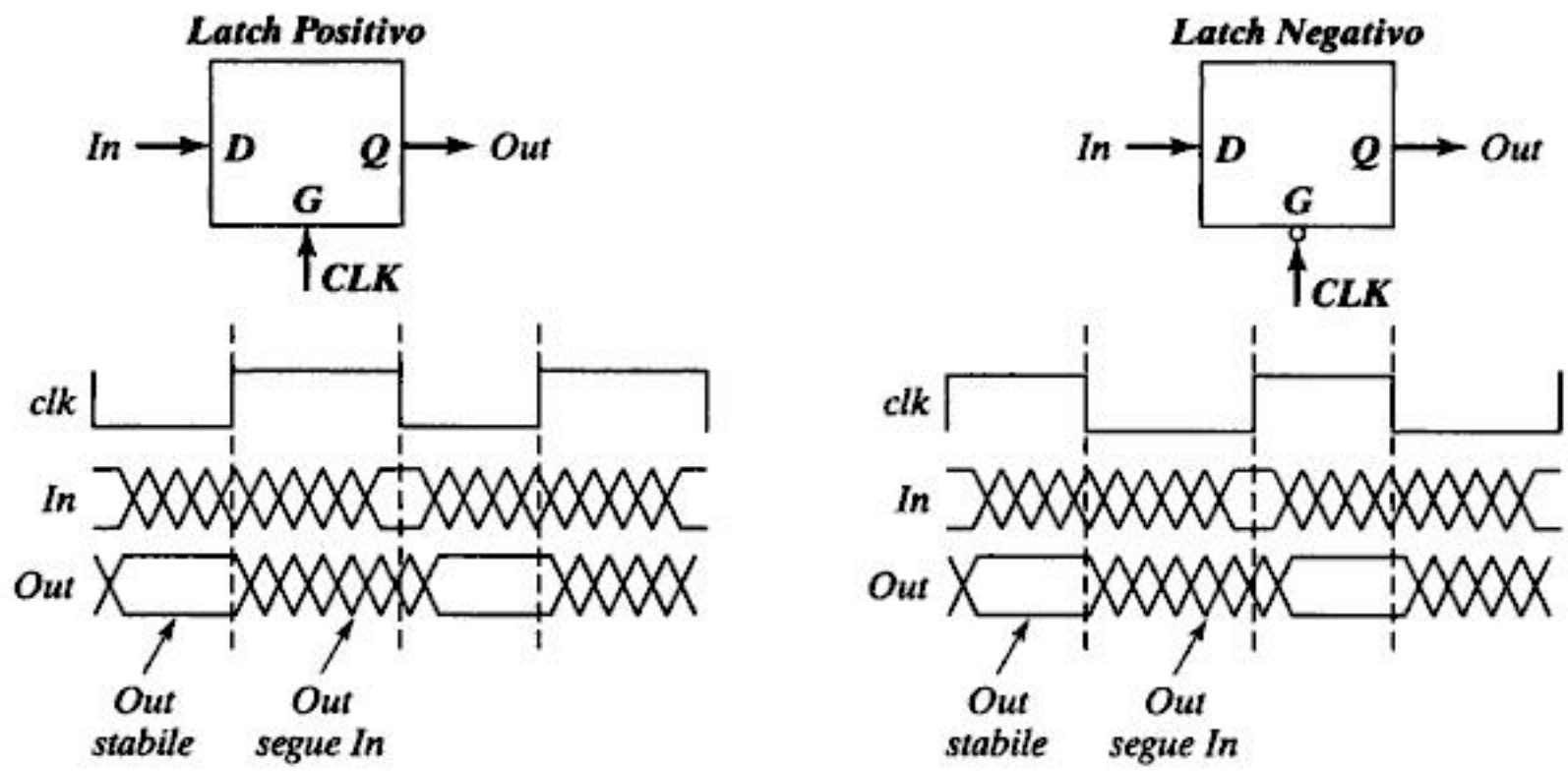


Figura 7-3 Temporizzazione di un latch positivo e di uno negativo.

La letteratura sui circuiti sequenziali è stata afflitta da definizioni ambigue dei differenti tipi di elementi in grado di memorizzare uno stato (cioè, registri, flip-flop e latch). Per evitare qualsiasi confusione, in questo libro ci conformeremo al seguente insieme di definizioni:

- Un elemento di memoria *edge-triggered* è detto *registro*.
- Un *latch* è un dispositivo *sensibile al livello* del clock.
- Qualsiasi elemento *bistabile*, formato dalla connessione incrociata di porte logiche, è detto *flip-flop*.[2]

7.2 Latch e registri statici

7.2.1 Il principio di bistabilità

Le memorie statiche sfruttano la retroazione positiva per creare un *circuito bistabile*, ovvero un circuito caratterizzato da due stati stabili che rappresentano 0 e 1. L'idea fondamentale è rappresentata nella Figura 7-4, che mostra due invertitori in cascata insieme alla caratteristica di trasferimento tipica di tale circuito. Sono riportate le caratteristiche del primo invertitore, cioè V_{o1} in funzione di V_{i1}, e del secondo invertitore (V_{o2} in funzione di V_{o1}). Quest'ultimo grafico è ruotato per sottolineare il fatto che $V_{i2} = V_{o1}$. Ipotizziamo ora che l'uscita del secondo invertitore V_{o2} sia connessa all'ingresso del primo V_{i1}, come mostrato con tratto punteggiato nella Figura 7-4a. Il circuito che ne risulta ha solamente tre possibili punti di lavoro (A, B e C), come dimostra la sovrapposizione delle due caratteristiche. Risulta quindi facile dimostrare l'importante affermazione che segue:

Quando il guadagno dell'invertitore nella regione di transizione è maggiore di uno, *A* e *B* sono gli unici punti operativi stabili e *C* è un punto operativo metastabile.

[2]Un registro edge-triggered è spesso indicato anche come flip-flop. In questo testo, il termine flip-flop è usato **unicamente** per indicare elementi bistabili.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

deviazione dal punto operativo relativamente grande si riduce fino a scomparire. Quindi la connessione incrociata di due invertitori dà origine a un circuito *bistabile*, cioè un circuito con due stati stabili, ciascuno corrispondente a uno stato logico. Il circuito può funzionare da memoria, immagazzinando un 1 o uno 0 (corrispondenti agli stati A e B).

Al fine di modificare lo stato memorizzato, dobbiamo essere in grado di portare il circuito dal punto operativo A a B e viceversa. Dato che la condizione necessaria per la stabilità è che il guadagno di anello G sia minore di uno, possiamo modificare lo stato A (o B) rendendolo temporaneamente instabile, portando cioè G a un valore maggiore di uno. Questo effetto è generalmente ottenuto applicando un impulso per l'innesco della commutazione a V_{i1} o V_{i2}. Ad esempio, ipotizziamo che il sistema sia nel punto A ($V_{i1} = 0$, $V_{i2} = 1$); forzare V_{i1} a 1 causa per un breve periodo l'accensione contemporanea di entrambi gli invertitori e un guadagno G maggiore di uno. La retroazione positiva rigenera l'effetto dell'impulso e il circuito si porta nell'altro stato (B, in questo caso). La durata dell'impulso deve essere di poco superiore al tempo di propagazione lungo l'anello, che è il doppio del tempo di propagazione medio degli invertitori.

Riassumendo, un circuito bistabile ha due stati stabili. In assenza di stimoli, il circuito permane in uno dei due stati (assumendo che la tensione di alimentazione sia sempre applicata al circuito) e quindi memorizza un valore. Un altro nome comunemente usato per indicare un circuito bistabile è *flip-flop*. Un flip-flop è di qualche utilità solo se esiste un modo per portare il circuito da uno stato all'altro. In generale esistono due diversi approcci per ottenere tale effetto:

- **Interrompere l'anello di retroazione.** Una volta che l'anello di retroazione è interrotto, si può facilmente scrivere un nuovo valore all'uscita Q. Tale latch è detto *a multiplexer*, dato che l'espressione logica che realizza un latch sincrono di questo tipo è identica all'equazione caratteristica di un multiplexer:

$$Q = \overline{Clk} \cdot Q + Clk \cdot In \tag{7.3}$$

  Questo approccio è il più comune nei latch odierni e quindi costituisce in nucleo fondamentale di questo paragrafo.
- **Contrastare l'anello di retroazione.** Applicando un segnale di innesco all'ingresso del flip-flop, è possibile forzare un nuovo valore nella cella, contrastando il valore memorizzato precedentemente. Affinché tale approccio funzioni è richiesto un dimensionamento accurato dei transistor nell'anello di retroazione e nella circuiteria di ingresso. Una rete di ingresso debole potrebbe essere inefficace nel contrastare una forte retroazione. Questo approccio era molto di moda all'inizio dell'era digitale, ma il suo indice di gradimento si è lentamente ridotto. Rimane comunque l'approccio dominante nel caso delle memorie globali statiche (che discuteremo con maggior completezza nel Capitolo 12) e ne daremo quindi una breve introduzione alla fine di questo capitolo.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

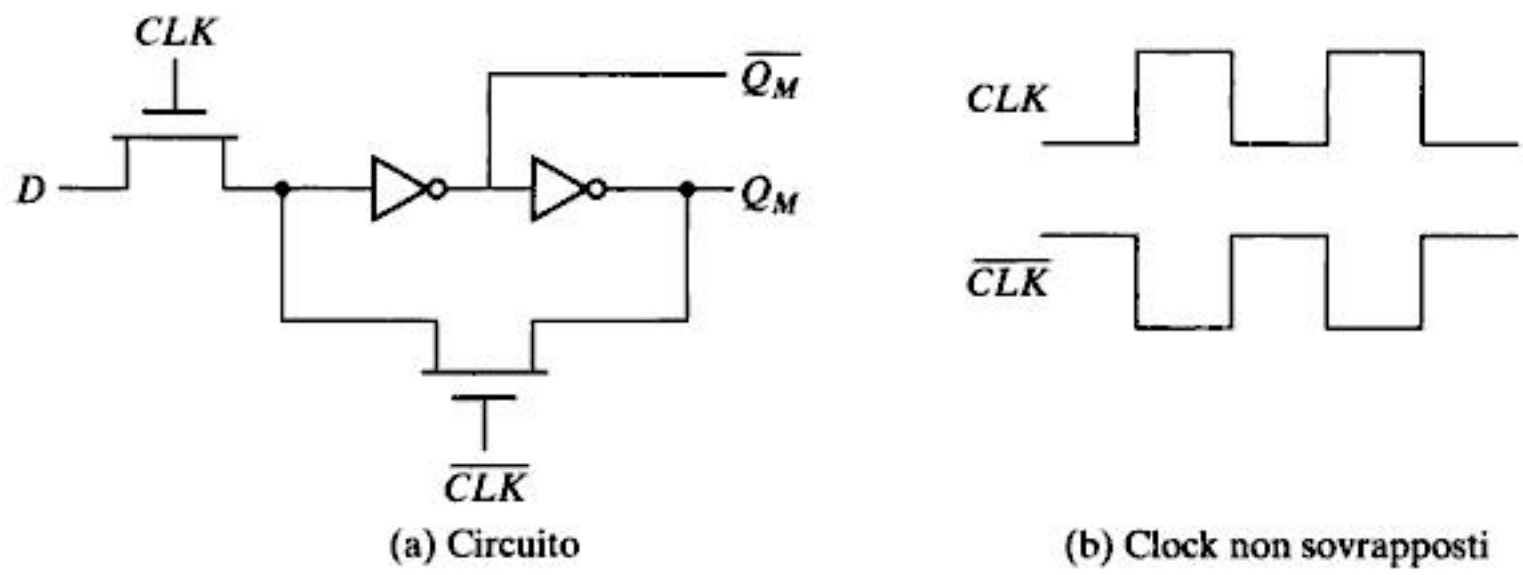


Figura 7-8 Latch NMOS basato su multiplexer che usa pass transistor a soli NMOS per il multiplexer.

È possibile ridurre il carico del clock a due transistor, realizzando il multiplexer con soli pass transistor NMOS, come si può vedere nella Figura 7-8. Quando *CLK* è alto, il latch campiona l'ingresso *D*, mentre un segnale di clock basso abilita l'anello di retroazione e pone il latch in modalità di memorizzazione. Sebbene sia attraente per la sua semplicità, l'uso di pass transistor NMOS implica la trasmissione all'ingresso del primo invertitore di una tensione di livello logico alto degradata ($V_{DD} - V_{Tn}$). Questo fatto ha conseguenze negative sia in termini di margini al rumore sia in termini di velocità di commutazione, specialmente nel caso di bassi valori di V_{DD} e V_{Tn} elevate. Inoltre il primo invertitore risente di un consumo statico di potenza non nullo, perché la sua tensione massima di ingresso è $V_{DD} - V_{Tn}$ e quindi il suo *pull-up* non è mai completamente spento.

7.2.3 Registri master–slave ed edge-triggered

L'approccio più comune per realizzare un *registro edge-triggered* è di usare una configurazione *master–slave*, come illustrato nella Figura 7-9. Il registro consiste della cascata di un latch negativo (stadio master) con uno positivo (stadio slave). In questo caso particolare usiamo latch a multiplexer, sebbene in generale sia possibile usare qualsiasi tipo di latch. Durante la fase bassa del clock, lo stadio master è trasparente e l'ingresso *D* è trasmesso all'uscita dello stadio master, Q_M. Durante questo periodo, lo stadio slave è in modalità di

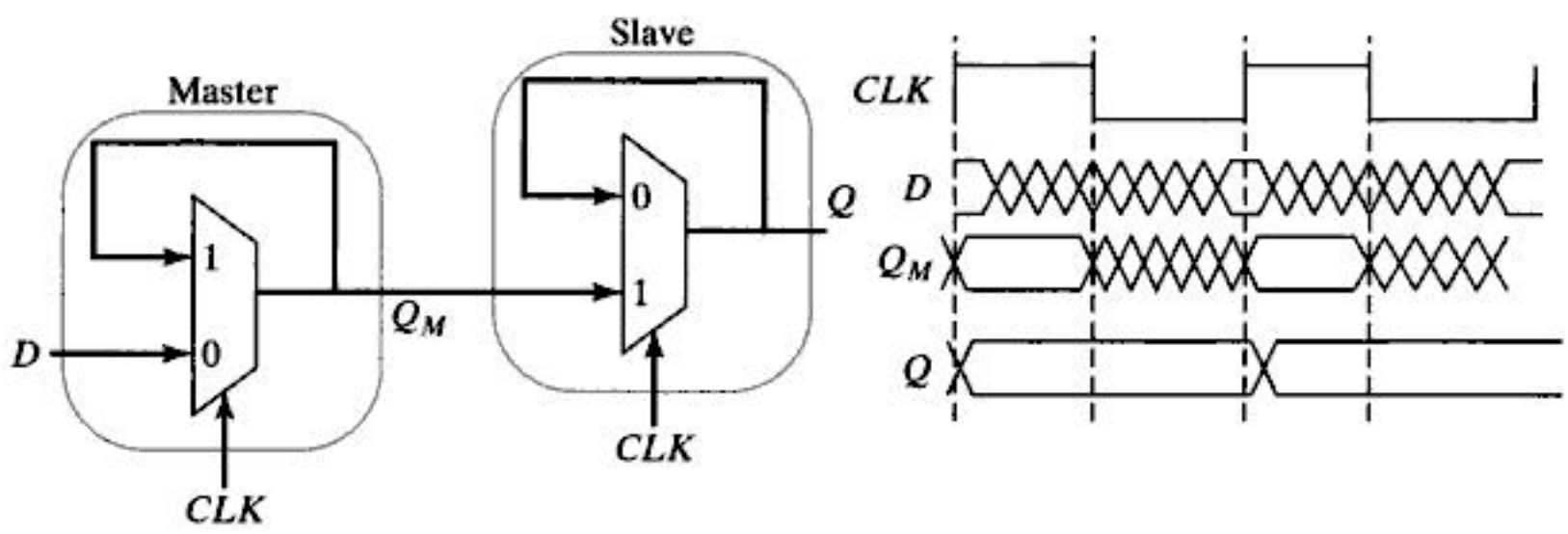


Figura 7-9 Registro positive edge-triggered in configurazione master–slave.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Il *tempo di setup* è il tempo prima del fronte di salita del clock per cui l'ingresso D deve essere valido. Ciò equivale a domandarsi per quanto tempo l'ingresso D deve essere stabile prima del fronte di salita del clock affinché nel nodo Q_M venga campionato correttamente il valore dell'ingresso. Per il registro a gate di trasmissione e multiplexer, l'ingresso D si deve propagare attraverso I_1, T_1, I_3 e I_2 prima della commutazione del clock. Questo assicura che le tensioni di entrambi i nodi del gate di trasmissione T_2 siano allo stesso valore. Altrimenti è possibile che la coppia incrociata I_2 e I_3 si stabilizzi a un valore sbagliato. Il tempo di setup è quindi $3 \times t_{pd_inv} + t_{pd_tx}$.

Il *tempo di propagazione* è il tempo necessario affinché il valore di Q_M si propaghi all'uscita Q. Notiamo che l'uscita di I_4 è valida prima del fronte di salita del clock, poiché abbiamo incluso il ritardo di I_2 nel tempo di setup. Quindi il tempo t_{c-q} è semplicemente il ritardo attraverso T_3 e I_6 ($t_{c-q} = t_{pd_tx} + t_{pd_inv}$).

Il *tempo di hold* rappresenta il tempo per cui l'ingresso deve essere mantenuto stabile dopo il fronte di salita del clock. In questo caso il gate di trasmissione T_1 si spegne quando il clock va alto. Dato che sia l'ingresso D sia CLK si propagano attraverso degli invertitori prima di raggiungere T_1, qualsiasi cambiamento dell'ingresso dopo che il clock è andato alto, non influenza l'uscita: quindi il tempo di hold è 0.

Esempio 7.1 Analisi temporale, utilizzando SPICE

Per ottenere il tempo di setup del registro utilizzando SPICE, sfasiamo progressivamente l'ingresso rispetto al fronte del clock, finché il dato campionato dal circuito è sbagliato. La Figura 7-11 mostra la simulazione del tempo di setup ipotizzando uno sfasamento di 210 ps e di 200 ps. Nel caso di 210 ps, viene campionato il valore corretto dell'ingresso D (in questo caso, l'uscita Q rimane al valore di V_{DD}). Per uno sfasamento di 200 ps, si propaga un valore sbagliato all'uscita, dato che Q commuta a 0. La

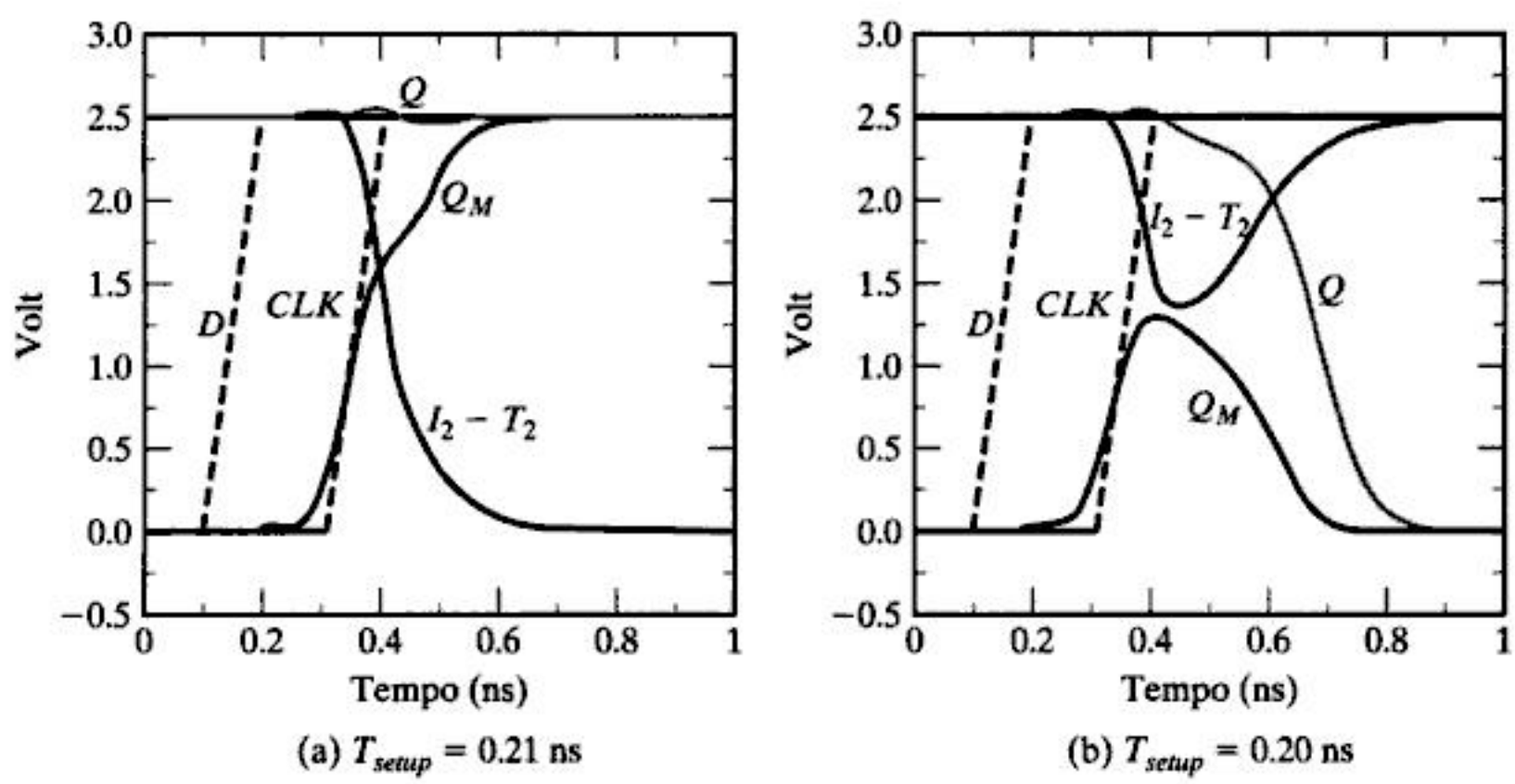


Figura 7-11 Simulazione del tempo di setup.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

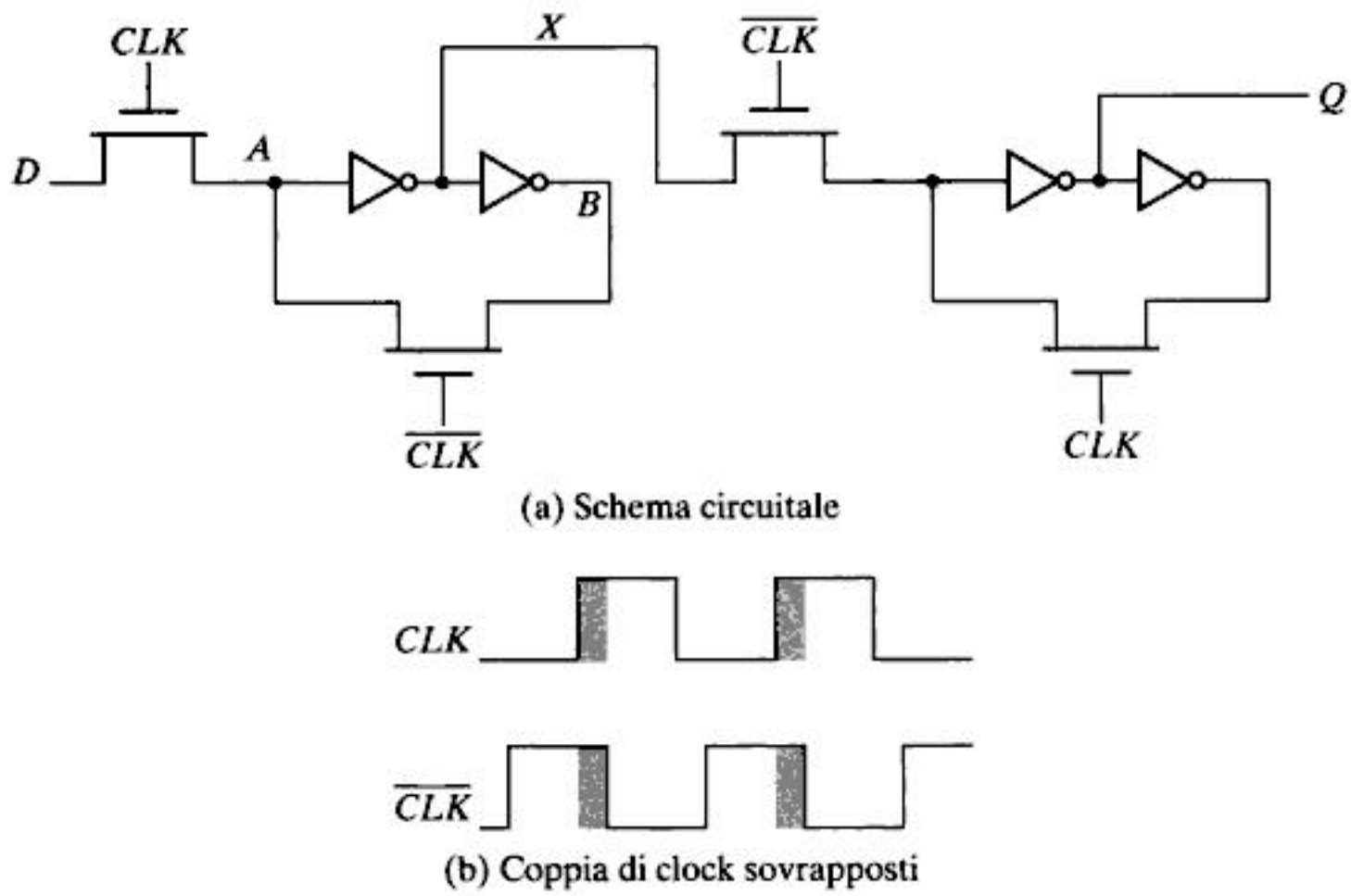


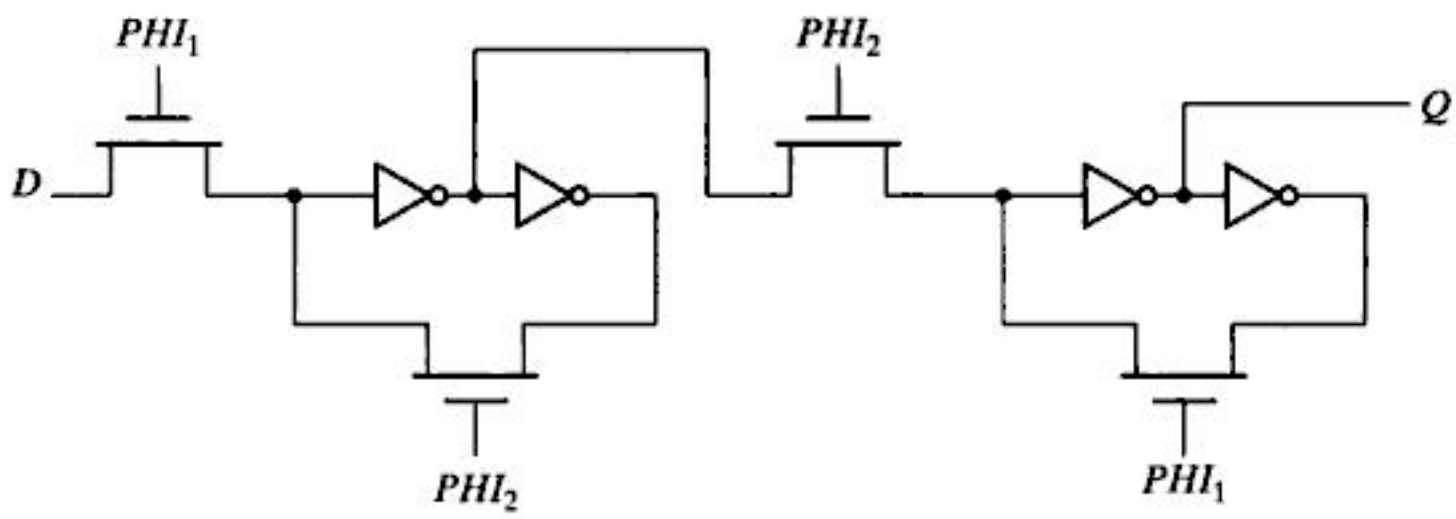
Figura 7-15 Registro master–slave a pass transistor con soli NMOS.

stor di campionamento conducono, formando un percorso diretto tra l'ingresso D e l'uscita Q. Di conseguenza, i dati in uscita possono cambiare in corrispondenza del fronte di salita del clock, fatto indesiderato per un registro negative edge-triggered. Questo fenomeno è noto come condizione di *corsa critica* (*race*) in cui il valore dell'uscita Q dipende dal fatto che l'ingresso D arrivi al nodo X prima o dopo il fronte di discesa di $\overline{CLK}$. Infine se il valore del nodo X viene campionato nello stato metastabile, l'uscita si porterà a un valore casuale determinato dal rumore del sistema.

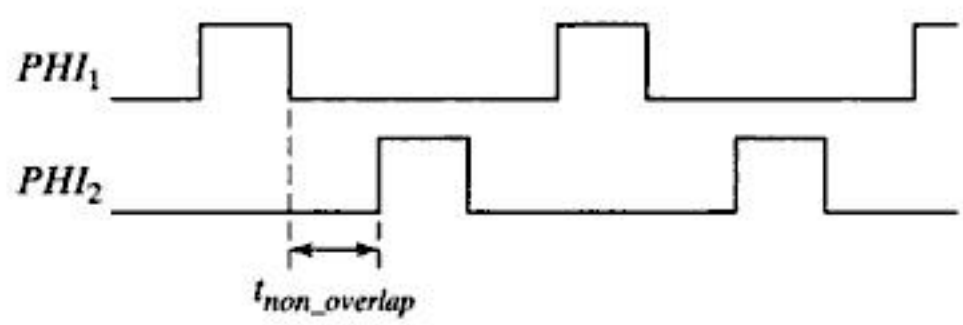
2. Il vantaggio principale di un registro a multiplexer è che l'anello di retroazione è aperto durante i periodi di campionamento e quindi il dimensionamento dei dispositivi non è critico rispetto alla funzionalità. Se esiste sovrapposizione tra CLK e $\overline{CLK}$ però, il nodo A può essere pilotato sia da D che da B, portandosi in uno stato indefinito.

Questi problemi possono essere evitati usando due *clock non sovrapposti* PHI_1 e PHI_2 (Figura 7-16) e assicurando che il tempo di non sovrapposizione t_{ns} sia abbastanza lungo da garantire che non si verifichi sovrapposizione, perfino in presenza dei ritardi dovuti alla distribuzione dei segnali di clock. Durante il tempo di non sovrapposizione, il flip-flop è in uno stato di alta impedenza: l'anello di retroazione è aperto, il guadagno di anello è nullo e l'ingresso è isolato. Se tale configurazione dura troppo a lungo, le correnti di perdita possono distruggere lo stato, da cui il nome di *pseudostatico*: il registro utilizza una combinazione degli approcci alla memorizzazione statici e dinamici, a seconda di quale sia lo stato dei clock.

Materiale protetto da copyright



(a) Circuito



(b) Clock a due fasi non sovrapposte

Figura 7-16 Registro *D* pseudostatico a due fasi.

Problema 7.2 Generazione di clock non sovrapposti

La Figura 7-17 mostra una possibile implementazione di un circuito per la generazione di un clock a due fasi non sovrapposte. Assumendo che ciascuna porta logica abbia un ritardo unitario, calcolare le relazioni temporali tra il clock di ingresso e le due fasi si uscita. Quanto vale il periodo di non sovrapposizione? Come si può aumentare tale intervallo in caso di necessità?

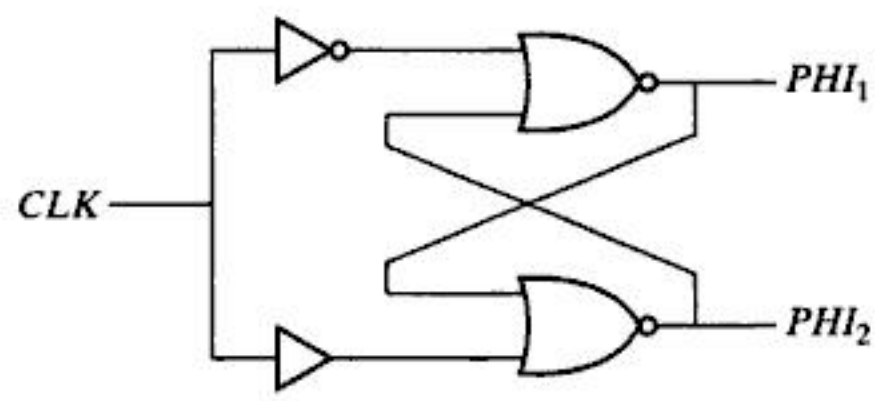


Figura 7-17 Circuito per la generazione di clock a due fasi non sovrapposte.

7.2.4 Latch statici a bassa tensione

La riduzione della tensione di alimentazione è un fattore fondamentale in circuiti a bassa potenza. Sfortunatamente, alcuni latch non funzionano correttamente a basse tensioni di alimentazione: per esempio, nel caso in cui le tensioni di soglia non vengano ridotte dello stesso fattore della tensione di alimentazione, usando pass transistor (per esempio, Figura 7-16) la degradazione della tensione associata al livello logico alto diviene più evidente. A tensioni di alimentazioni molto basse, il segnale di ingresso all'invertitore potrebbe non superare la ten-

Materiale protetto da copyright

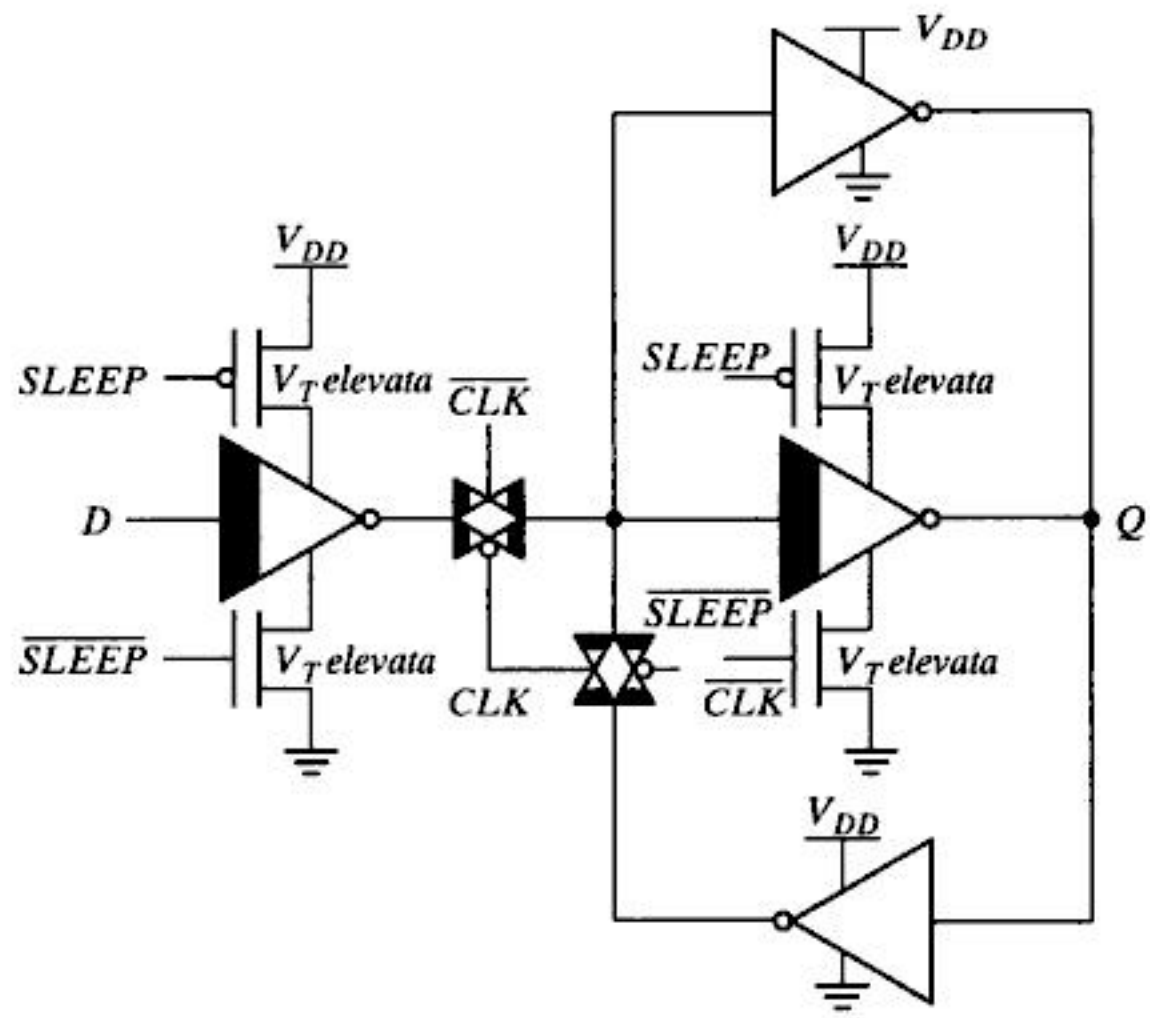


Figura 7-18 Soluzione al problema delle correnti di perdita usando dispositivi CMOS a soglia multipla.

sione di soglia logica, compromettendo il funzionamento corretto. Perfino usando un gate di trasmissione, le prestazioni degradano in modo inaccettabile per tensioni di alimentazione ridotte.

La riduzione della tensione di alimentazione richiede quindi anche la riduzione della soglia dei dispositivi. Questo però ha la conseguenza negativa di far aumentare esponenzialmente le correnti di perdita in sottosoglia e il relativo consumo di potenza (come abbiamo visto nel Capitolo 6). Quando si accede ai registri regolarmente, l'energia consumata per compensare le correnti di perdita è tipicamente insignificante rispetto all'energia spesa per la commutazione dei nodi; usando clock condizionati, però, è possibile che i registri rimangano disabilitati per lunghi periodi e l'energia associata alle correnti di perdita sia significativa. Sono state studiate diverse soluzioni per risolvere il problema delle perdite durante i periodi di inattività. Uno degli approcci possibili sfrutta l'uso di transistor con soglie diverse (*multiple-threshold CMOS, MTCMOS*), come mostrato nella Figura 7-18 [Mutoh95], nel caso del latch negativo. Gli invertitori e i gate di trasmissione ombreggiati sono realizzati con dispositivi a bassa tensione di soglia. Gli invertitori con tali dispositivi sono connessi a massa e all'alimentazione attraverso dispositivi a tensione di soglia elevata, in modo da eliminare le correnti di perdita. Durante il normale funzionamento, i dispositivi con tensione di soglia elevata sono accesi e, quando il clock è basso, l'ingresso *D* viene campionato e propagato all'uscita. Il latch è in fase di memorizzazione quando il clock è alto: il gate di trasmissione in retroazione conduce e quindi la retroazione è abilitata. Per immagazzinare lo stato quando il latch è *disabilitato* (*sleep*), è presente un invertitore aggiuntivo, in parallelo con quello a tensione di soglia ridotta. In modalità di memoria, i dispositivi a tensione di soglia elevata, in

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

dell'invertitore M_1–M_2. Una volta ottenuta tale condizione, la retroazione positiva fa sì che il flip-flop inverta il suo stato. Questo vincolo ci costringe ad aumentare le dimensioni dei transistor M_5, M_6, M_7 e M_8. La combinazione dei transistor M_4, M_7 e M_8 forma un invertitore a rapporto. Ipotizziamo che la coppia di invertitori incrociati sia progettata in modo che la soglia logica dell'invertitore V_M sia pari a $V_{DD}/2$. Nella tecnologia CMOS da 0.25 μm, abbiamo scelto le dimensioni seguenti: $(W/L)_{M_1} = (W/L)_{M_3} = (0.5\ \mu m/0.25\ \mu m)$, e $(W/L)_{M_2} = (W/L)_{M_4} = (1.5\ \mu m/0.25\ \mu m)$. Assumendo $Q = 0$, possiamo ora ricavare le dimensioni minime di M_5, M_6, M_7 e M_8 che garantiscono la possibilità di far commutare il dispositivo.

Per fare commutare il latch dallo stato $Q = 0$ allo stato $Q = 1$, è essenziale che il livello logico basso dell'invertitore a rapporto pseudo-NMOS (M_5–M_6–M_2) sia minore della soglia di commutazione dell'invertitore M_3–M_4, che è pari a $V_{DD}/2$. È ragionevole ipotizzare che finché $V_{\overline{Q}} > V_M$, V_Q è pari a 0 e il gate del transistor M_2 è a massa. Si può ricavare un limite inferiore per le dimensioni dei transistor uguagliando le correnti dell'invertitore per $V_{\overline{Q}} = V_{DD}/2$, come mostrato nell'Eq. (7.4) (dove si è ignorata la modulazione della lunghezza del canale). Le correnti sono quelle di saturazione, poiché $V_S = V_{DD} = 2.5$ V e $V_M = 1.25$ V. Ipotizziamo che M_5 e M_6 abbiano la stessa dimensione e che $W/L_{5\text{–}6}$ indichi il fattore di forma equivalente della serie dei due dispositivi; in tali condizioni la rete di pull-down può essere rappresentata da un singolo transistor $M_{5\text{–}6}$, la cui lunghezza è doppia rispetto a quella di un dispositivo singolo:

$$\begin{aligned} k'_n\left(\frac{W}{L}\right)_{5-6}\left((V_{DD}-V_{Tn})V_{DSATn}-\frac{V_{DSATn}^2}{2}\right) \\ = -k'_p\left(\frac{W}{L}\right)_2\left((-V_{DD}-V_{Tp})V_{DSATp}-\frac{V_{DSATp}^2}{2}\right) \end{aligned} \tag{7.4}$$

Usando i parametri del processo da 0.25 μm, l'Eq. (7.4) porta alla condizione $(W/L)_{M_{5\text{–}6}} \geq 2.26$. Questo significa che il fattore di forma per M_5 e M_6 deve essere almeno 4.5. La Figura 7-22a mostra la caratteristica statica di $V_{\overline{Q}}$ in funzione della dimensione del singolo dispositivo M_5 o M_6. Notiamo che una dimensione maggiore di 3 è già sufficiente a portare la tensione di $\overline{Q}$ fino alla tensione di soglia logica dell'invertitore. La discrepanza tra l'analisi manuale e la simulazione deriva dagli effetti del secondo ordine, quali modulazione della lunghezza di canale e DIBL. La Figura 7-22b riporta la risposta al transitorio per diverse dimensioni dei dispositivi e conferma che un fattore di forma W/L maggiore di 3 per i singoli dispositivi è necessario per contrastare la retroazione e innescare la commutazione dello stato del latch.

7.3 Latch e registri dinamici

Un circuito sequenziale statico si basa sul principio di bistabilità, ottenuto mediante una coppia di invertitori mutuamente retroazionati e in grado quindi di memorizzare due valori

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

trollati (per esempio un'unità di elaborazione ad alte prestazioni progettata in modo completamente dedicato), dovrebbero essere pseudostatici o statici. Questa considerazione è valida per tutti i latch e registri esaminati in questo paragrafo.

7.3.2 C²MOS: una soluzione insensibile al clock-skew

Il registro C²MOS

La Figura 7-26 mostra un ingegnoso registro positive edge-triggered, basato su un'architettura master–slave insensibile alla sovrapposizione dei clock. Questo circuito è chiamato *registro C²MOS* (*Clocked CMOS*) [Suzuki73] e opera in due fasi:

1. $CLK = 0$ ($\overline{CLK} = 1$): il primo invertitore tristate è acceso e il master funziona come un normale invertitore campionando la versione invertita di D sul nodo interno X; in tale configurazione il master è detto in modalità di valutazione. Nel frattempo, la sezione slave è in modalità di alta impedenza o memorizzazione. Entrambi i transistor M_7 e M_8 sono spenti, disaccoppiando l'ingresso dall'uscita. L'uscita Q mantiene il suo valore precedente immagazzinato sulla capacità di uscita C_{L2}.
2. I ruoli si scambiano quando $CLK = 1$: lo stadio master è in modalità di memorizzazione (M_3-M_4 spenti), mentre il secondo stadio è in valutazione (M_7–M_8 accesi). Il valore immagazzinato in C_{L1} si propaga al nodo di uscita attraverso lo stadio slave, che si comporta da invertitore.

Il circuito nel suo complesso si comporta come un registro positive edge-triggered master–slave ed è molto simile al registro a gate di trasmissione presentato prima. Vi è però una differenza importante:

Un registro C²MOS è insensibile alla sovrapposizione dei clock CLK–$\overline{CLK}$, se i tempi di salita e discesa di questi ultimi sono sufficientemente piccoli.

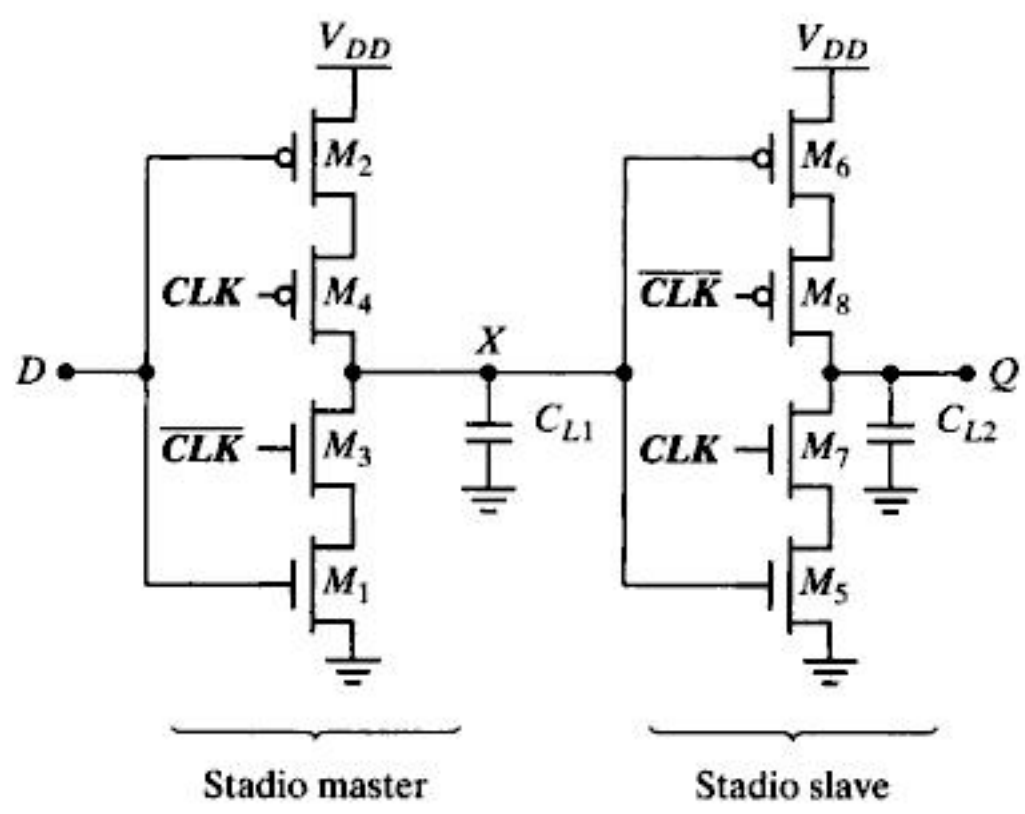


Figura 7-26 Registro C²MOS master–slave positive edge-triggered.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

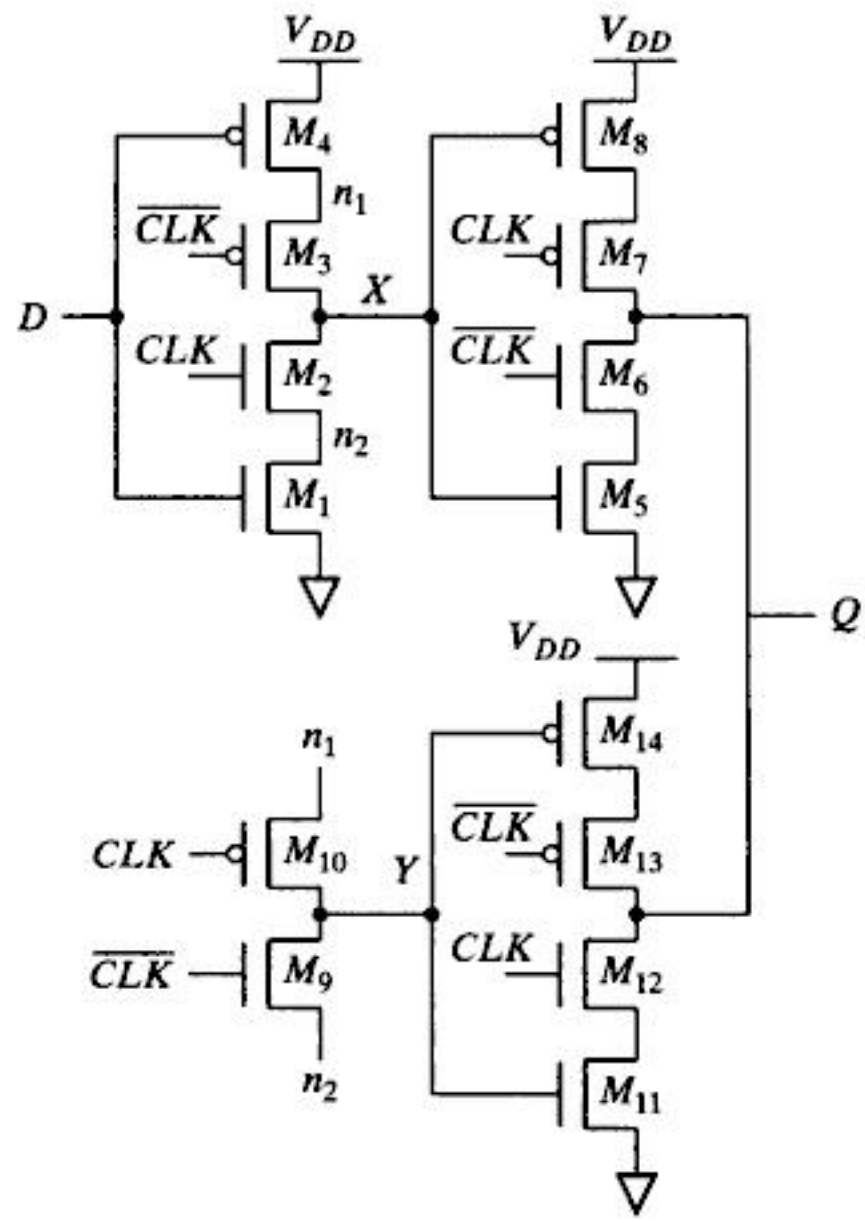


Figura 7-29 Registro C²MOS a doppio fronte.

l'ingresso D invertito sul nodo Y. Notate che i dispositivi M_1 e M_4 sono condivisi da entrambi i master, riducendo il carico all'ingresso D. Sul fronte di salita, il latch slave inferiore conduce e forza la versione invertita di Y sul nodo Q. I dati sono quindi campionati su entrambi i fronti. Notiamo che i latch slave operano in modo complementare: uno solo è acceso durante ogni fase del clock.

Problema 7.5 Registri a fronti duali

Determinare il modo in cui l'impiego di registri a fronti duali influenza il consumo di potenza nella rete di distribuzione del clock.

7.3.3 I registri a fase di clock singola

Negli schemi di temporizzazione a due fasi descritti fin qui, è necessario fare attenzione nella distribuzione delle due fasi, in modo che la sovrapposizione sia minima. Mentre l'approccio C²MOS rappresenta una soluzione in grado di tollerare un certo livello di sfasamento temporale, è possibile progettare i registri in modo che utilizzino una fase di clock singola. I *registri a fase di clock singola (True Single-Phase Clocked Register*, TSPC), proposti da Yuan e Svensson, usano un **unico clock** [Yuan89]. La struttura fondamentale di un latch a fase singola, positivo e negativo, è mostrata nella Figura 7-30. Per il latch positivo, quando CLK è alto, il latch è in modalità trasparente e corrisponde alla cascata di due invertitori; il latch è

Materiale protetto da copyright

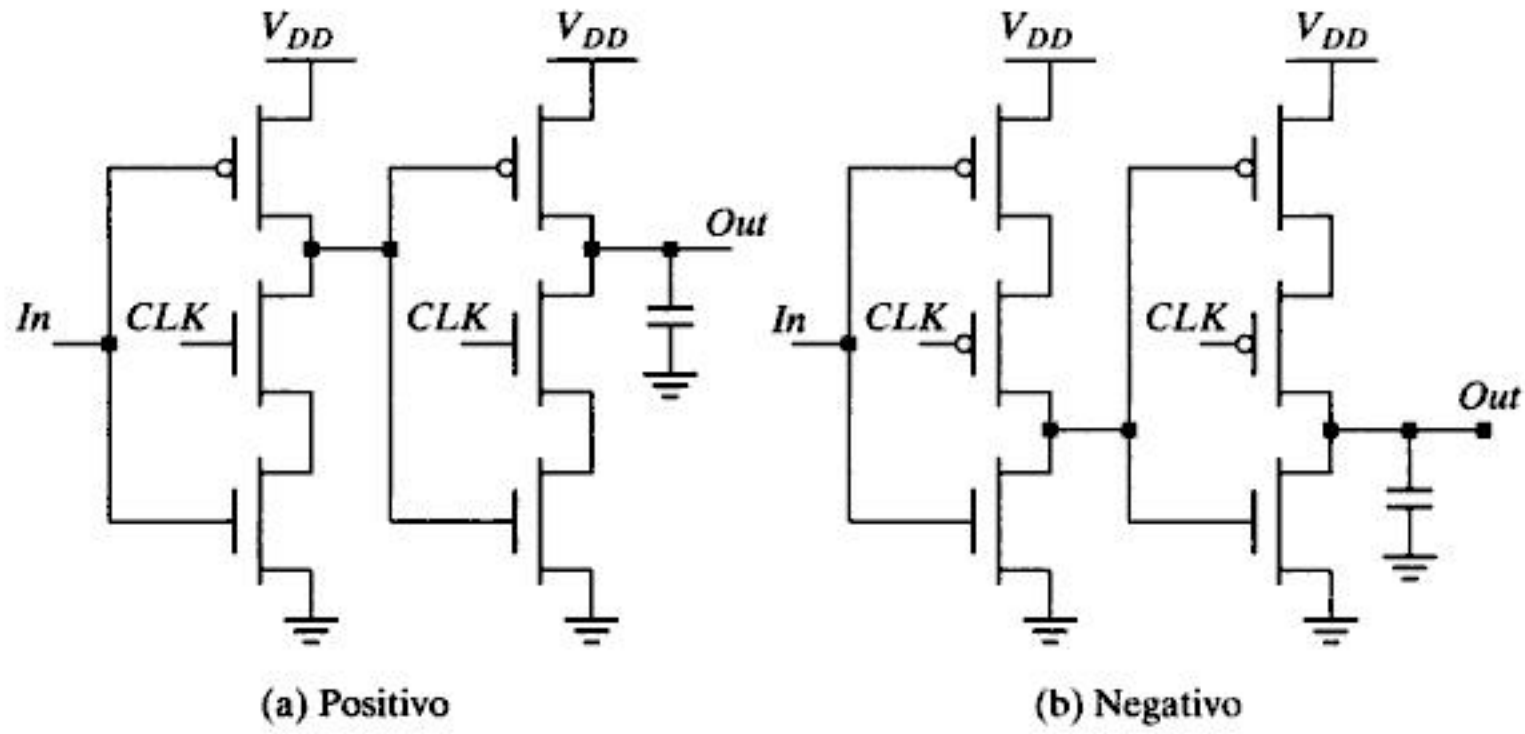


Figura 7-30 Latch a fase di clock singola; (a) positivo e (b) negativo.

non invertente e propaga l'ingresso fino all'uscita. D'altra parte, quando $CLK = 0$, entrambi gli invertitori sono disabilitati e il latch è in modalità di memorizzazione. Solamente le reti di pull-up sono ancora attive, mentre i circuiti di pull-down sono disattivati. In conseguenza dell'approccio a doppio stadio, nessun segnale si può propagare dall'ingresso fino all'uscita in questa modalità. Un registro può essere realizzato ponendo in cascata un latch positivo e uno negativo. Il carico per il segnale di clock è analogo a quello di un registro a gate di trasmissione convenzionale oppure di un registro C^2MOS. Il vantaggio principale consiste nell'uso di un singolo clock, mentre lo svantaggio risiede nel leggero aumento del numero di transistor (ora ne occorrono 12).

Bisogna notare che un circuito dinamico come quello della Figura 7-30 va usato con cautela. Quando il clock è basso (per il latch positivo), il nodo di uscita può essere flottante ed è esposto all'accoppiamento con altri segnali. Inoltre si possono verificare fenomeni di ridistribuzione della carica se il nodo di uscita pilota un gate di trasmissione. I nodi dinamici dovrebbero essere isolati con l'aiuto di invertitori statici, oppure resi pseudostatici al fine di migliorarne l'immunità al rumore.

Così come molte altre famiglie di latch, i registri TSPC offrono un vantaggio aggiuntivo che non abbiamo ancora messo in evidenza: la possibilità di integrare delle funzioni logiche nel latch. Questo permette di ridurre l'aumento dei ritardi associato all'introduzione del latch. La Figura 7-31a mette in evidenza l'approccio fondamentale per includere la logica, mentre la Figura 7-31b mostra un esempio di latch positivo che implementa la AND di In_1 e In_2 oltre che svolgere la funzione di latch. Mentre il tempo di setup di questo latch è aumentato rispetto a quello della Figura 7-30, la prestazione globale del circuito logico (cioè il periodo minimo di clock del circuito sequenziale) è migliorato: l'aumento del tempo di setup è tipicamente inferiore al tempo di propagazione di una porta AND. La tecnica di includere la logica nei latch è stata usata estesamente nella progettazione del microprocessore Alpha EV4 DEC [Dobberpuhl92], così come in molti altri processori ad alte prestazioni.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

7.4 Altri tipi di registri*

7.4.1 Registri con clock a impulsi

Finora abbiamo usato la configurazione master–slave per realizzare un registro edge-triggered. Un approccio completamente diverso per ottenere un registro è quello di usare *segnali impulsivi*, in cui l'idea fondamentale è quella di generare un breve impulso nell'intorno del fronte di salita (o di discesa) del clock. Tale impulso può poi essere utilizzato come segnale di sincronizzazione per un latch (per esempio, Figura 7-35a), abilitando il campionamento del segnale solo per un breve intervallo. Le condizioni di corsa critica vengono quindi evitate mantenendo il tempo di apertura del latch (cioè, il periodo in cui è trasparente) molto breve. La combinazione del circuito che genera l'impulso e il latch forma un registro positive edge-triggered.

La Figura 7-35b mostra un esempio di circuito che genera un breve impulso di tensione in corrispondenza di ogni fronte di salita del clock [Kozu96]. Quando $CLK = 0$, il nodo X è carico a V_{DD} (M_N è spento poiché $CLKG$ è basso). In corrispondenza del fronte di salita del clock, c'è un breve periodo di tempo per cui entrambi gli ingressi della porta AND sono alti, causando la commutazione verso il livello alto di $CLKG$. Questa commutazione a sua volta attiva M_N, scaricando X e di conseguenza anche $CLKG$ (Figura 7-35c). La durata dell'impulso è determinata dal ritardo della porta AND e dei due invertitori. Notiamo che esiste anche un ritardo tra il fronte di salita del clock di ingresso (CLK) e il clock impulsivo ($CLKG$), uguale alla somma del ritardo della porta AND e dei due invertitori. Se ogni registro

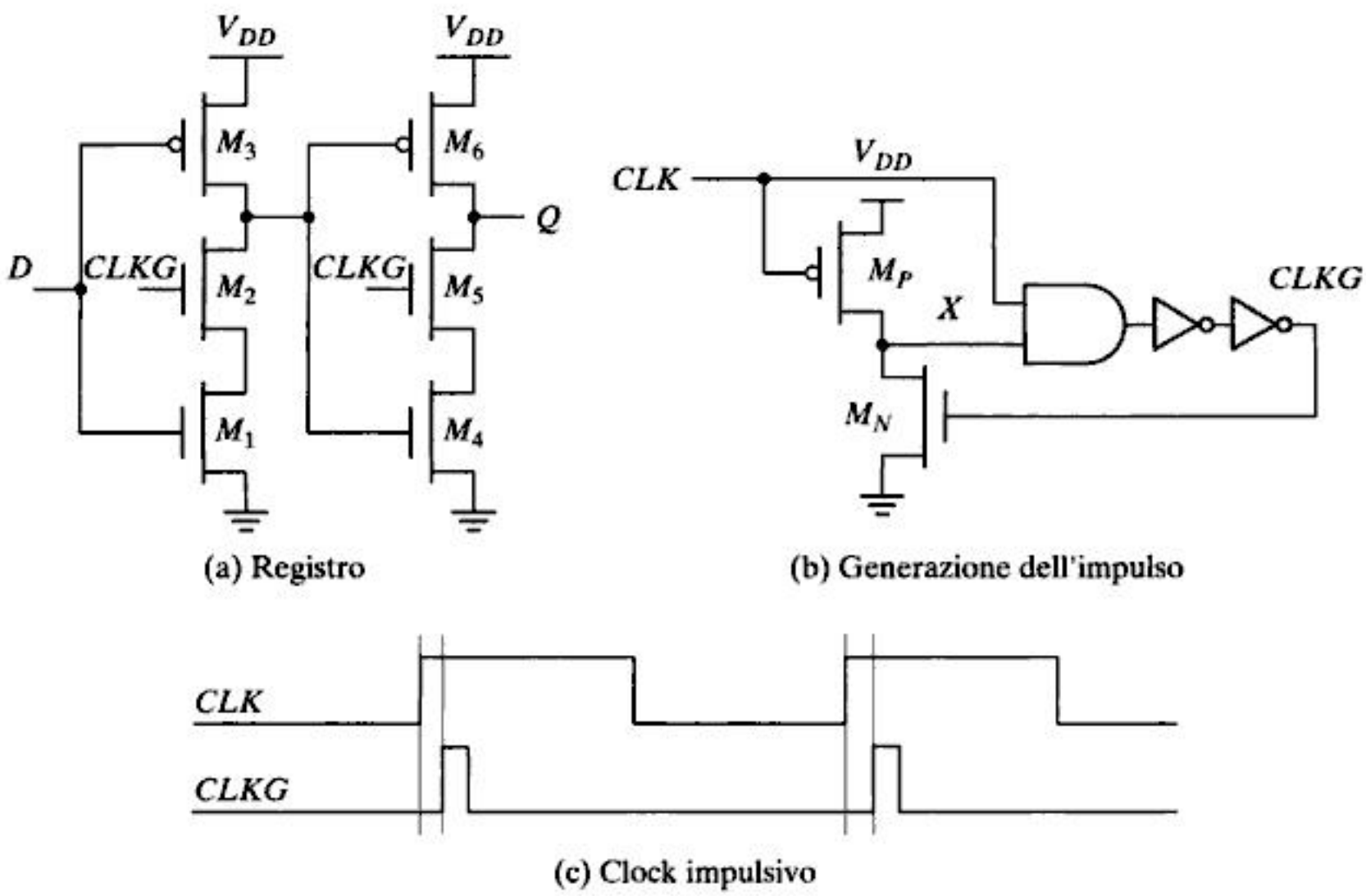


Figura 7-35 Registro e generazione della temporizzazione in un latch impulsivo basato su TSPC.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

propagarsi fino all'uscita Q. In questo particolare circuito, il tempo di setup può essere negativo: è il caso in cui il periodo di trasparenza è maggiore del ritardo dall'ingresso all'uscita. Si tratta di una caratteristica interessante, perché implica che i dati possono arrivare al registro perfino dopo la commutazione a livello alto del clock. Quindi è possibile prendere in prestito del tempo dal ciclo precedente.

Esempio 7.5 Tempo di setup del registro impulsivo

Il registro impulsivo della Figura 7-36 è trasparente durante la sovrapposizione (1–1) di CLK e $\overline{CLKD}$. Di conseguenza, il dato di ingresso può in effetti cambiare anche dopo il fronte di salita del clock, ottenendo un tempo di setup negativo (Figura 7-37). L'ingresso D commuta a livello basso dopo il fronte di salita del clock e commuta a livello alto dopo il fronte di discesa di $\overline{CLKD}$ (cioè, durante il periodo di trasparenza). Osserviamo in che modo l'uscita segue l'ingresso: l'uscita Q assume il valore corretto di V_{DD} finché l'ingresso D converge correttamente al suo valore in anticipo rispetto al fronte di discesa di $\overline{CLKD}$. Quando si sfrutta il tempo di setup negativo, non vi è alcuna garanzia sul comportamento monotono dell'uscita; cioè l'uscita può avere commutazioni multiple nell'intorno del fronte di salita e quindi il registro non dovrebbe essere usato per pilotare porte logiche dinamiche o per generare il clock per altri registri.

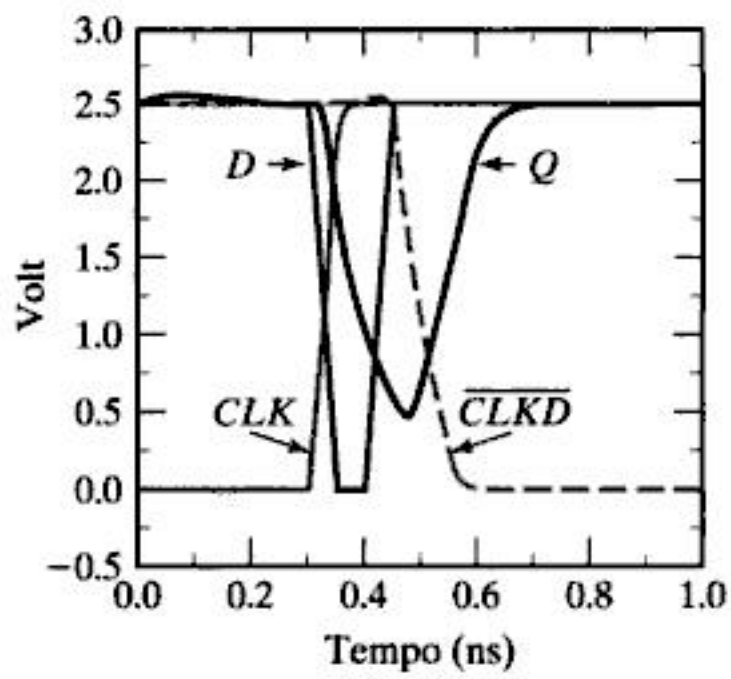


Figura 7-37 Simulazione che mostra un tempo di setup negativo in un registro impulsivo.

Problema 7.6 Conversione di un registro impulsivo in un registro impulsivo condizionato

Modificare il circuito della Figura 7-36 in modo da aggiungere un ingresso di *abilitazione*. L'obiettivo è di trasformare il registro in un registro condizionato, che memorizza solo quando il segnale di abilitazione è attivato.

Materiale protetto da copyright

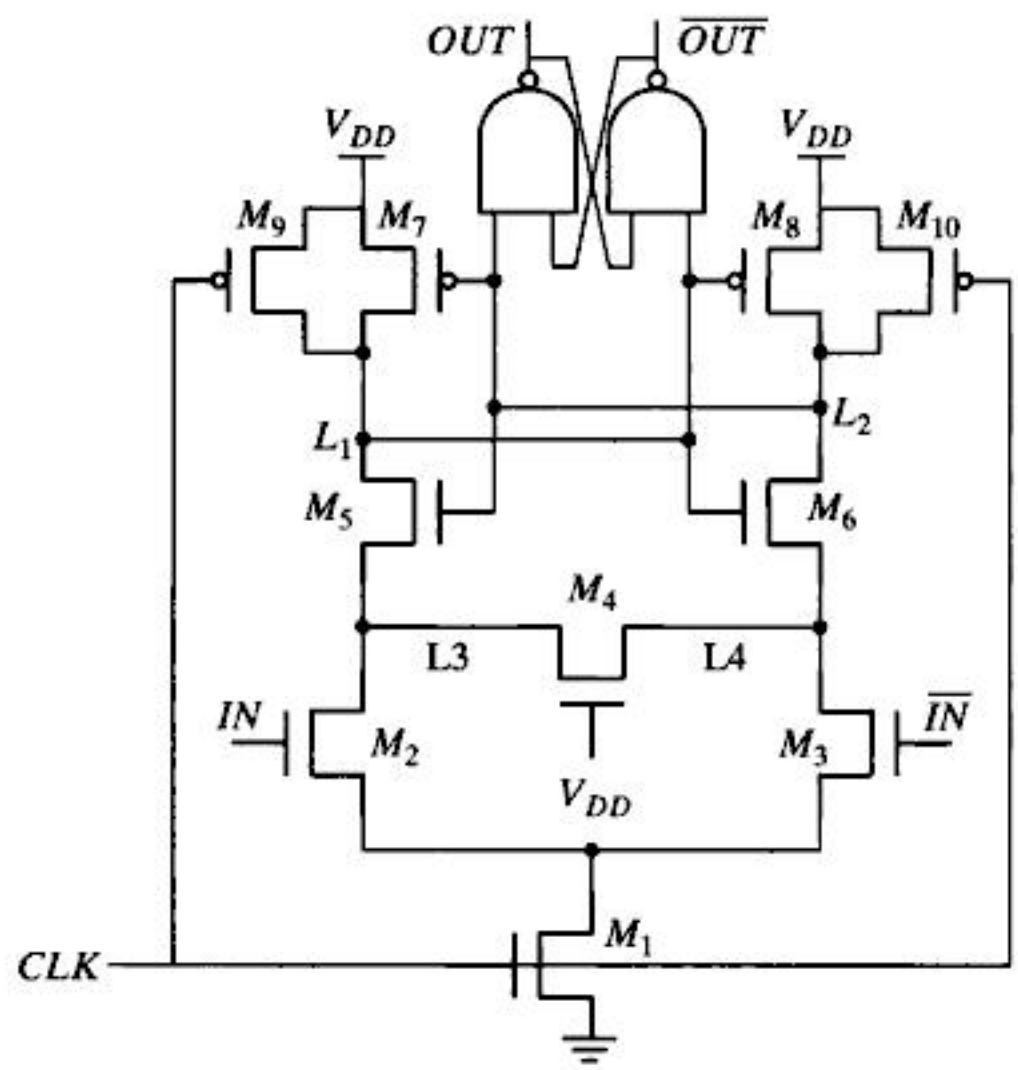


Figura 7-38 Registro positive edge-triggered a sense amplifier.

7.4.2 Registri a Sense-amplifier

Oltre agli approcci *master–slave* e *impulsivi* per implementare un registro edge-triggered, può essere usata una terza tecnica basata su *sense amplifier*, come si può vedere nella Figura 7-38 [Montanaro96].[3] I sense amplifier ricevono in ingresso piccoli segnali e li amplificano fino a generare un'escursione logica completa. Sono utilizzati estesamente nei nuclei delle memorie e nei bus a escursione ridotta, al fine di migliorare le prestazioni o ridurre il consumo di potenza. Vi sono molte tecniche per realizzare questi amplificatori: un approccio molto comune consiste nell'usare la retroazione, per esempio, attraverso un insieme di invertitori incrociati. Il circuito mostrato nella Figura 7-38 usa un preamplificatore dinamico che campiona il segnale di ingresso differenziale in corrispondenza al fronte di salita del clock. Le uscite di tale amplificatore fungono da ingressi per un flip-flop SR a NAND che memorizza il dato e garantisce che le uscite differenziali non effettuino più di una commutazione a ogni ciclo di clock. In questo circuito non è necessario che gli ingressi differenziali abbiano un'escursione completa.

In nucleo del preamplificatore consiste di un invertitore incrociato (M_5–M_8), le cui uscite (L_1 e L_2) vengono precaricate durante la fase bassa del clock, utilizzando i dispositivi M_9 e M_{10}. Di conseguenza i transistor PMOS M_7 e M_8 vengono spenti e il flip-flop a NAND mantiene lo stato precedente. Il transistor M_1 è simile a un interruttore di valutazione in un circuito dinamico e viene spento per garantire che gli ingressi differenziali non influiscano sulle uscite durante la fase bassa del clock. Sul fronte di salita del clock, il transistor di valu-

[3]In un certo senso, questi registri basati su sense amplifier operano in modo simile ai registri impulsivi: il primo stadio genera un impulso che abilita il secondo stadio a funzionare da latch.

Materiale protetto da copyright

tazione si accende, la coppia differenziale di ingresso (M_2 e M_3) è abilitata e la differenza tra i segnali di ingresso è amplificata sui nodi di uscita L_1 e L_2. La coppia di invertitori (M_5-M_7 e M_6-M_8) commuta a uno dei suoi due stati stabili, a seconda dei valori in ingresso. Per esempio, se *IN* è 1, L_1 è scaricato a 0 e L_2 rimane a V_{DD}. Grazie all'amplificazione del primo stadio, non è necessario che gli ingressi arrivino fino a V_{DD}, consentendo l'uso di segnali a escursione ridotta sulle piste di ingresso.

Il transistor M_4 è usato per fornire un percorso DC dal nodo L_3 o L_4verso massa per le correnti di perdita. Questo è necessario nel caso in cui gli ingressi commutino dopo il fronte di salita di *CLK*, lasciando L_3 o L_4 in uno stato di alta impedenza con memorizzato un valore logico basso. Senza il percorso per le correnti di perdita, il nodo può essere ricaricato e di conseguenza il latch può commutare stato prima del prossimo fronte di salita di *CLK*. Questo fenomeno può essere compreso meglio graficamente, come dimostra la Figura 7-39.

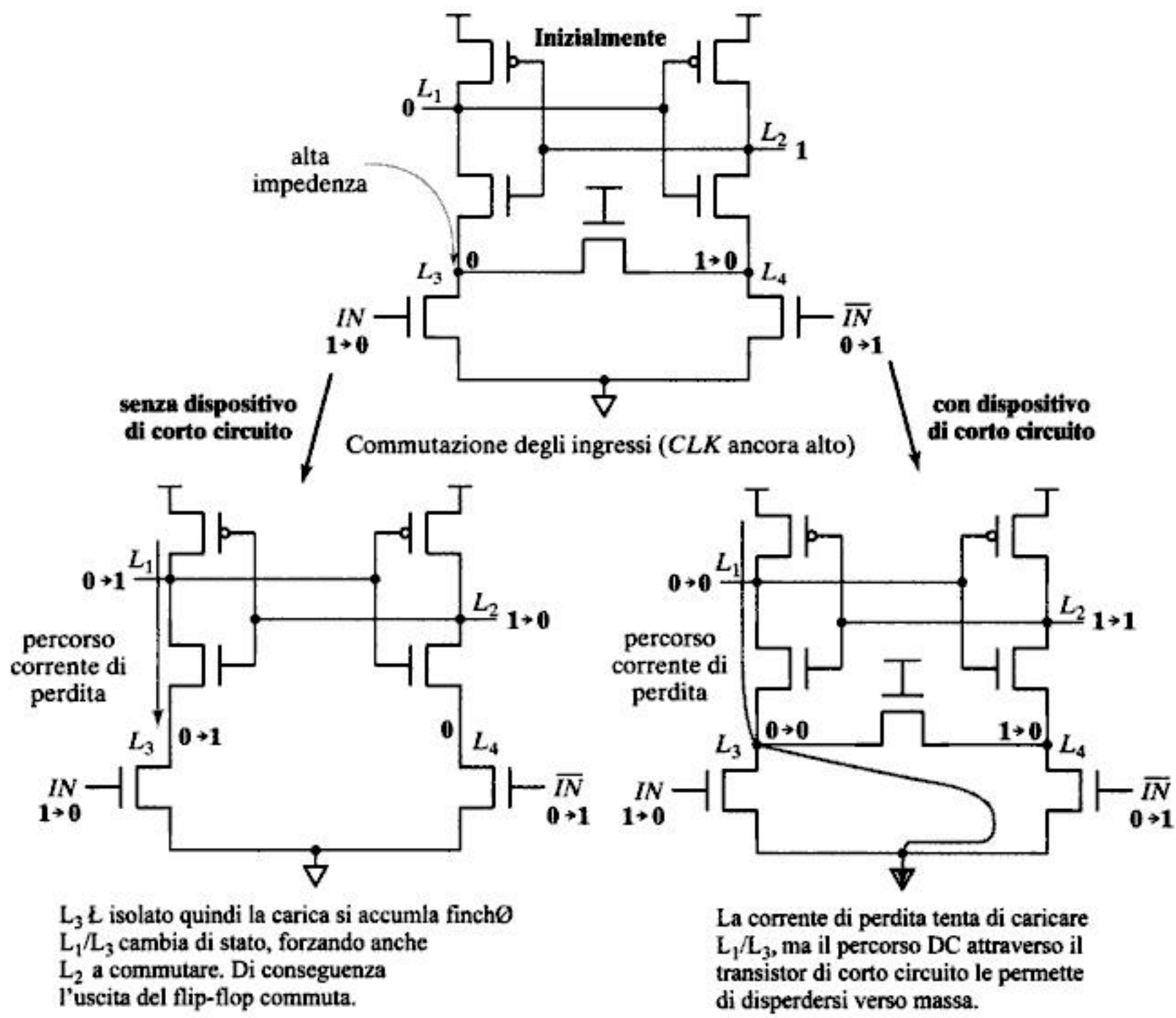


Figura 7-39 La necessità del transistor di corto circuito M_4 per il corretto funzionamwnto del latch a sense amplifier.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

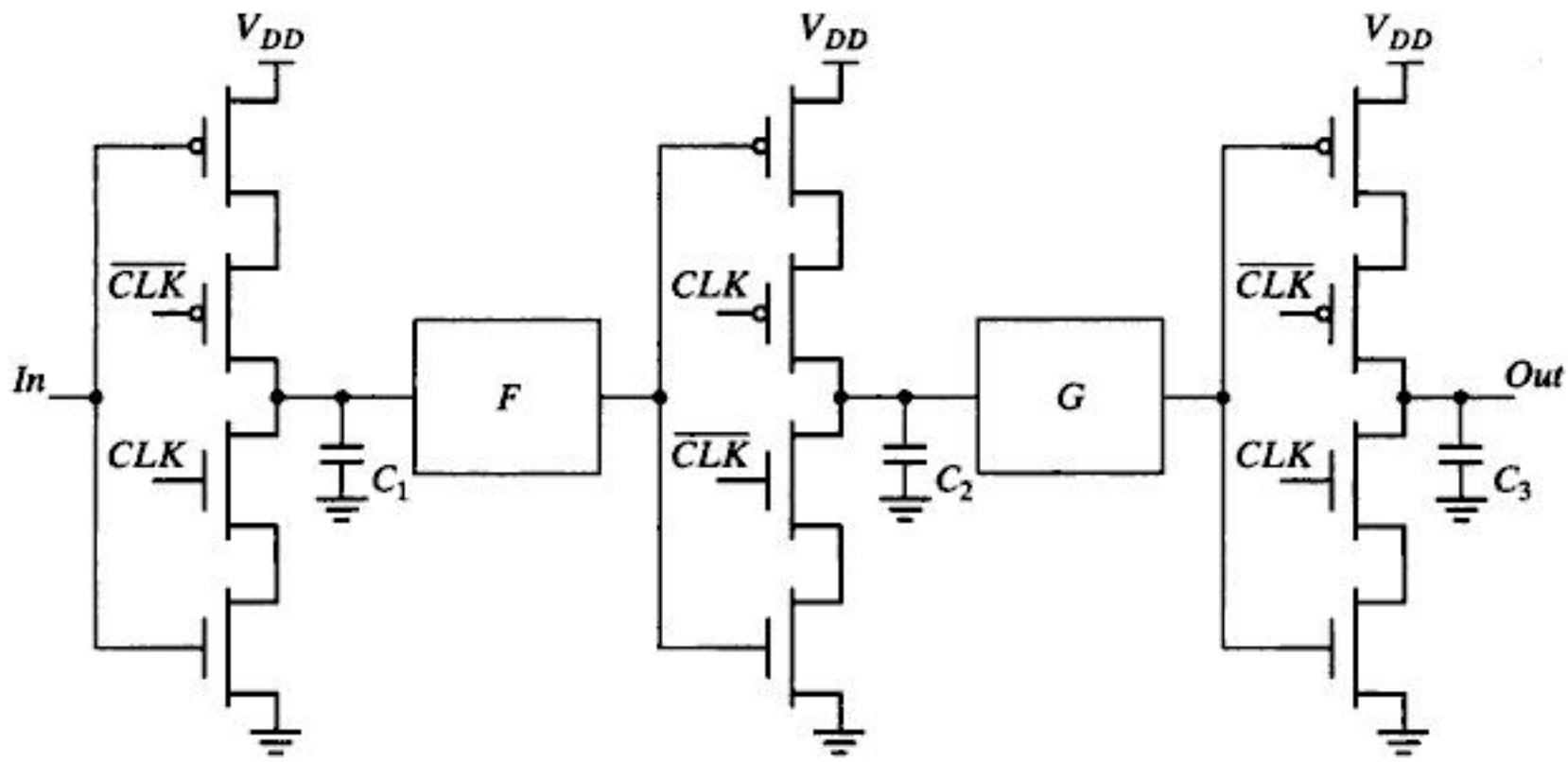


Figura 7-42 Percorso di segnale pipeline, utilizzando latch C²MOS.

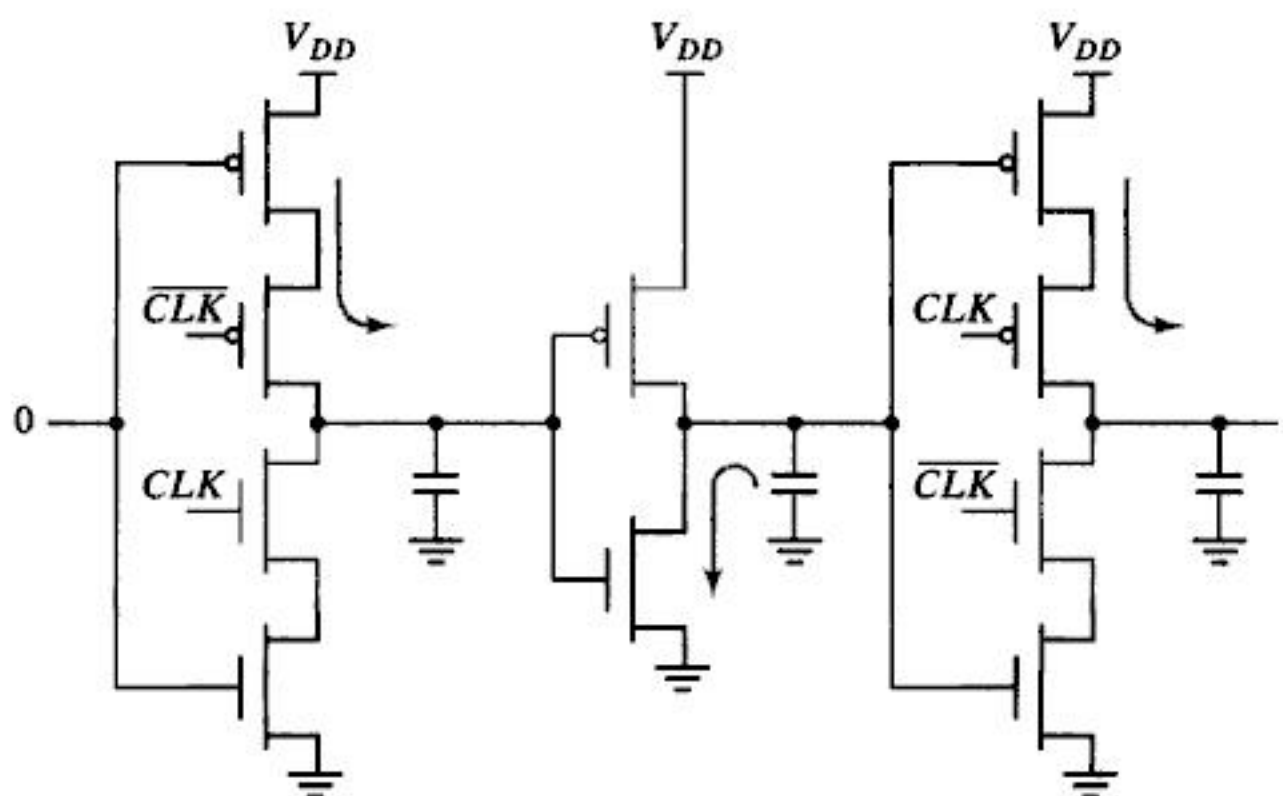


Figura 7-43 Potenziale condizione di corsa critica durante la sovrapposizione (0–0) in un sistema C²MOS.

C²MOS si semplificano a sole reti di pull-up (vedi Figura 7-27). In queste condizioni, un segnale può propagarsi da uno stadio al successivo solo se la funzione logica *F* è invertente, come mostrato nella Figura 7-43, dove *F* è rappresentata per semplicità da un singolo invertitore CMOS statico. Considerazioni del tutto simili sono valide per la sovrapposizione (1–1).

Sulla base di questo principio è stata ideata una struttura logica chiamata NORA–CMOS [Gonçalves83], che combina registri C²MOS e funzioni logiche dinamiche NORA in pipeline. Ciascun modulo consiste di un blocco di logica combinatoria che può essere un misto tra logica dinamica e statica, seguito da un latch C²MOS. Reti logiche e latch sono controllati dal clock in modo che per lo stesso valore logico del clock siano entrambi in valuta-

Materiale protetto da copyright

Tabella 7-2 Funzionamento di moduli logici NORA.

	blocco *CLK*		blocco $\overline{CLK}$	
	Logica	**Latch**	**Logica**	**Latch**
$CLK = 0$	Precarica	Memoria	Valutazione	Valutazione
$CLK = 1$	Valutazione	Valutazione	Precarica	Memoria

zione o in memorizzazione (precarica). Un blocco in fase di valutazione durante $CLK = 1$ è detto *modulo CLK*, mentre nel caso inverso è detto *modulo* $\overline{CLK}$. Un esempio di entrambi i moduli è mostrato nella Figura 7-44. Il modo corretto di sincronizzare il latch e la rete logica dinamica in entrambi i moduli è riassunto nella Tabella 7-2.

Un percorso di segnale in architettura NORA è formato dalla cascata di moduli *CLK* e $\overline{CLK}$ alternati. Mentre una classe di moduli è in precarica con il corrispondente latch di uscita in memorizzazione (mantenendo il valore di uscita precedente), l'altra classe è in valutazione. I dati vengono trasmessi in modo pipeline da modulo a modulo. L'architettura NORA offre ai progettisti un'ampia gamma di scelte progettuali. Logiche dinamica e statica possono essere mischiate liberamente e blocchi dinamici sia *CLK* che $\overline{CLK}$ possono essere usati in cascata o in configurazione pipeline. Sebbene questo stile logico permetta di evitare l'invertitore aggiuntivo richiesto nella logica domino CMOS, vi sono molte regole che devono essere rispettate per ottenere funzioni affidabili e prive di condizioni di corsa critica. In seguito a tale complessità, l'utilizzo delle architetture NORA è stato di fatto limitato solo ad applicazioni con prestazioni molto spinte.

7.6 Altri circuiti sequenziali

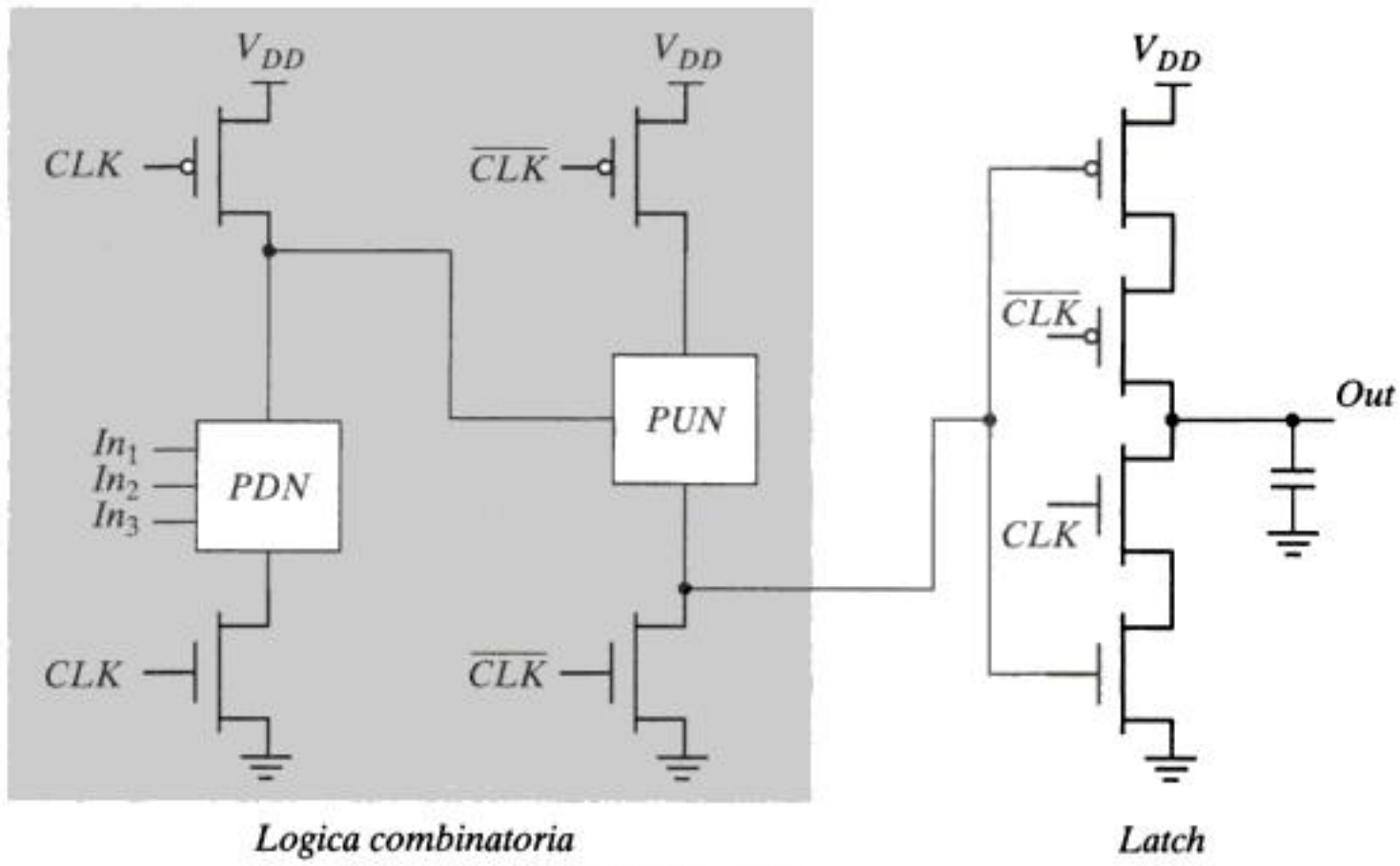
Nei paragrafi precedenti ci siamo concentrati su un unico tipo di elemento sequenziale: il latch (e il suo derivato, il registro). La proprietà più importante di questo circuito è il fatto che possiede due stati stabili e per questo è chiamato *bistabile*. Quello bistabile non è però l'unico elemento sequenziale di interesse: esistono altri circuiti rigenerativi catalogati come *astabili* e *monostabili*. I primi funzionano come oscillatori e possono, per esempio, essere utilizzati per creare dei generatori di clock integrati sul chip. I secondi funzionano come generatori di impulso e sono anche chiamati *circuiti a colpo unico* (*one-shot circuits*). Un altro circuito rigenerativo interessante è il *trigger di Schmitt*. Questo componente ha l'interessante proprietà di mostrare isteresi nella caratteristica statica: la sua soglia logica è variabile e dipende dal tipo di transizione (da basso ad alto o da alto a basso). Questa caratteristica peculiare può dimostrarsi di aiuto in ambienti rumorosi.

7.6.1 Il trigger di Schmitt

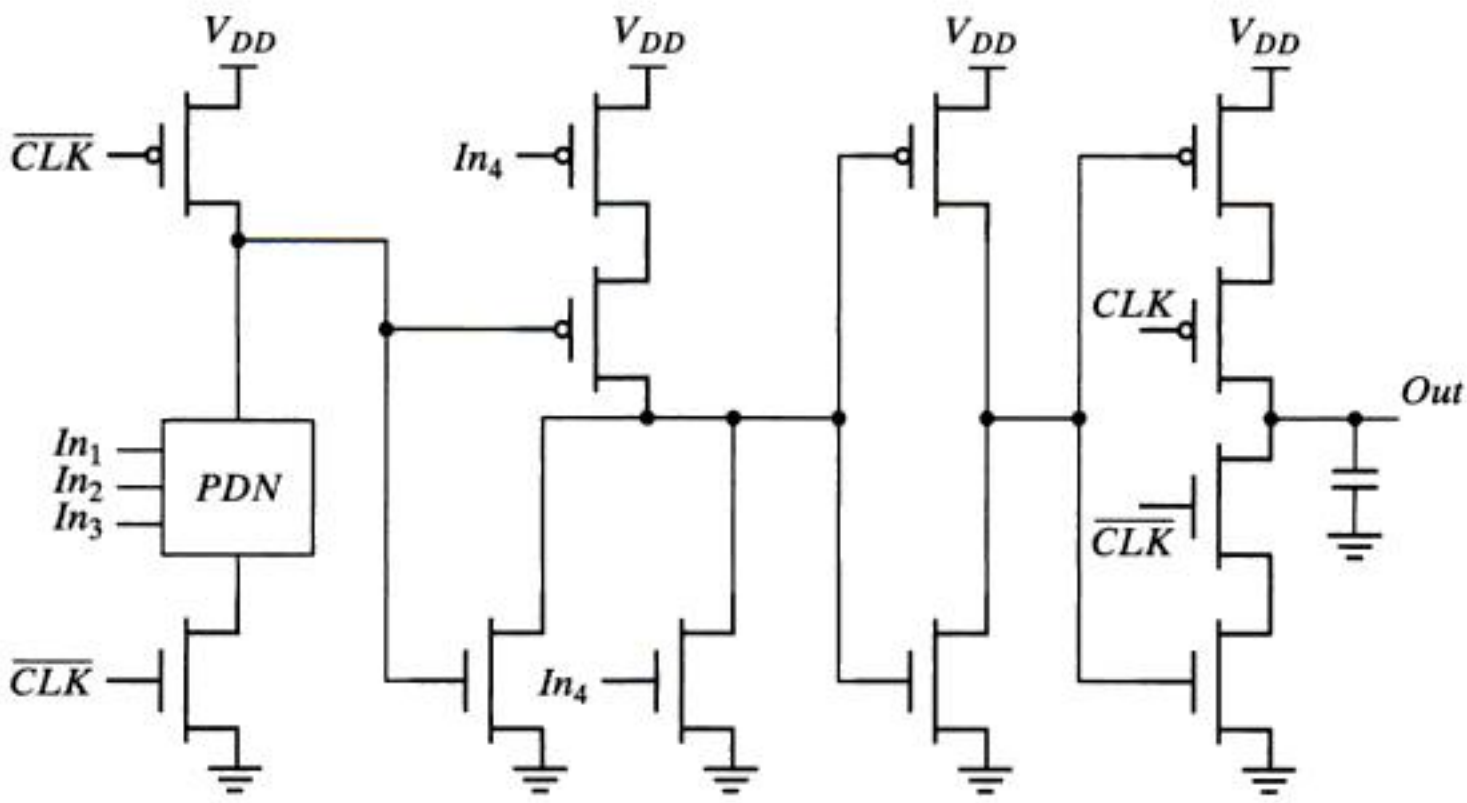
Definizione

Un *trigger di Schmitt* [Schmitt38] è un dispositivo con due importanti proprietà:

Materiale protetto da copyright



(a) *Modulo CLK*



(b) *Modulo* $\overline{CLK}$

Figura 7-44 Esempi di moduli NORA–CMOS.

1. Risponde a un ingresso lentamente variabile con una **transizione rapida in uscita**.
2. La caratteristica di trasferimento ha due *soglie logiche differenti quando il segnale di ingresso commuta verso l'alto oppure verso il basso*. Tutto ciò è illustrato nella Figura 7-45, che riporta la tipica caratteristica di trasferimento di un trigger di Schmitt e il suo simbolo. Le soglie logiche per la transizione basso-alto e alto-basso sono indicate rispettivamente con V_{M+} e V_{M-}; la *tensione di isteresi* è definita come la differenza tra queste due quantità.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

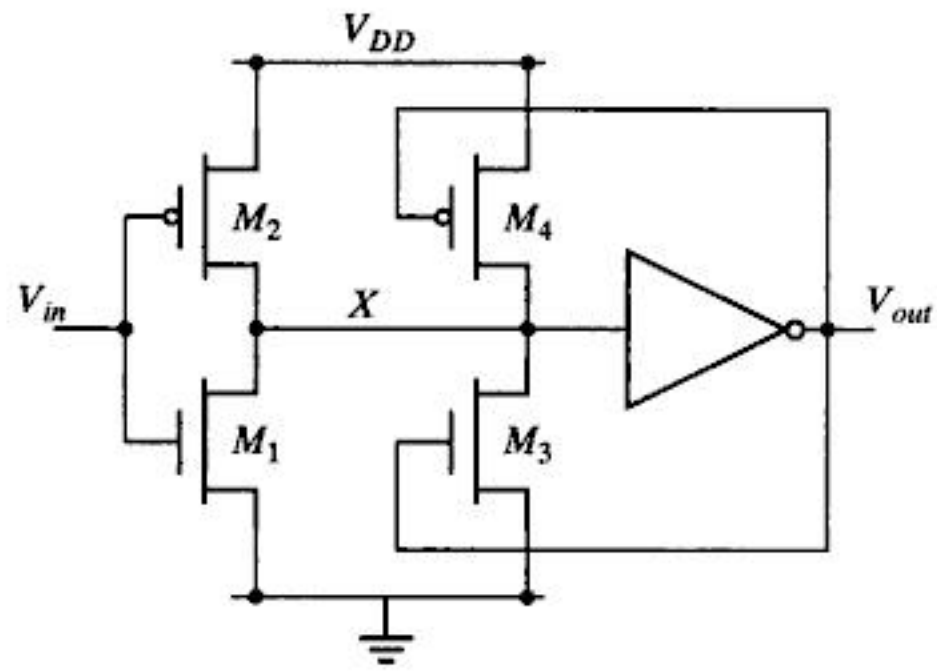


Figura 7-47 Un trigger di Schmitt CMOS.

un invertitore formato dal parallelo di due transistor PMOS (M_2 e M_4) come rete di pull-up e da un unico transistor NMOS (M_1) come rete di pull-down. Questa configurazione modifica il rapporto effettivo dei transistor dell'invertitore a $k_{M1}/(k_{M2} + k_{M4})$, che sposta la soglia logica verso l'alto.

Una volta che l'invertitore commuta, l'anello di retroazione spegne M_4, e il dispositivo M_3 viene attivato. Questo dispositivo di pull-down aggiuntivo, accelera la transizione e quindi produce un segnale di uscita pulito con commutazioni ripide.

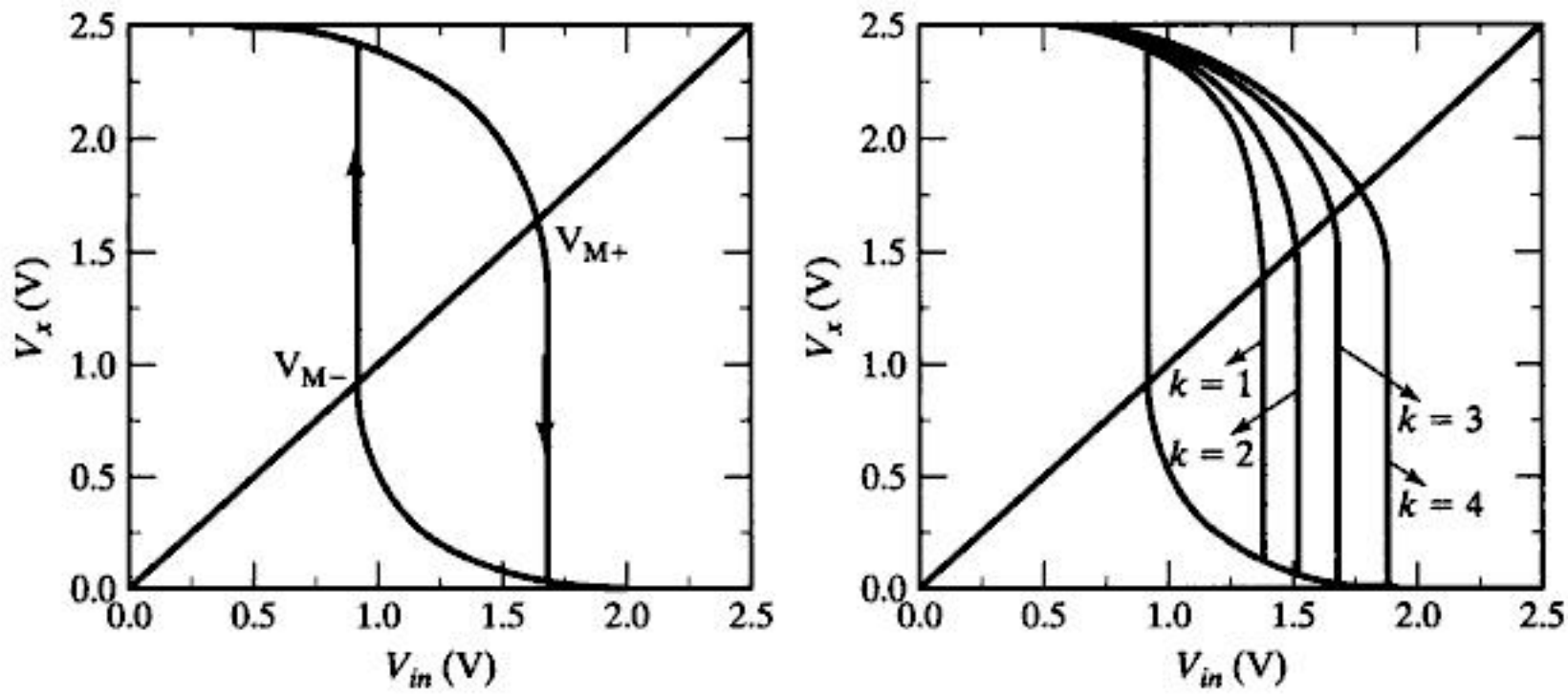
Un comportamento analogo può essere osservato nel caso di una commutazione alto-basso. In questo caso, la rete di pull-down è inizialmente formata dal parallelo di M_1 e M_3, mentre la rete di pull-up è formata da M_2. Questo riduce il valore della soglia logica a V_{M-}.

Esempio 7.6 Trigger di Schmitt CMOS

Consideriamo il trigger di Schmitt della Figura 7-47, con M_1 e M_2 di dimensione rispettivamente 1 μm/0.25 μm, e 3 μm/0.25 μm. L'invertitore è progettato in modo da avere la soglia logica nell'intorno di $V_{DD}/2$ (= 1.25 V). La Figura 7-48a mostra la simulazione del trigger di Schmitt assumendo che i dispositivi M_3 e M_4 abbiano rispettivamente dimensioni pari a 0.5 μm/0.25 μm e 1.5 μm/0.25 μm. Com'è evidente dal grafico, il circuito mostra isteresi. La soglia di commutazione per la transizione alto-basso ($V_{M-} = 0.9$ V) è minore di $V_{DD}/2$, mentre quella per la transizione basso-alto ($V_{M+} = 1.6$ V) è maggiore di $V_{DD}/2$.

È possibile spostare la soglia di commutazione, cambiando le dimensioni di M_3 e M_4. Per esempio, per modificare quella della transizione basso-alto, dobbiamo variare le dimensioni del dispositivo PMOS. La soglia della transizione alto-basso rimane costante mantenendo la larghezza del dispositivo M_3 a 0.5 μm. La larghezza del dispositivo M_4 viene modificata al valore $k \times 0.5$ μm: la Figura 7-48b dimostra che la tensione di soglia aumenta all'aumentare del fattore k.

Materiale protetto da copyright



(a) Caratteristica di trasferimento con isteresi

(b) Effetto della variazione del rapporto del PMOS M_4. La larghezza è $k \times 0.5$ μm.

Figura 7-48 ùSimulazioni del trigger di Schmitt.

Problema 7.7 Un trigger di Schmitt CMOS alternativo

Un altro trigger di Schmitt CMOS è mostrato nella Figura 7-49. Esaminare il funzionamento del circuito logico e derivare le espressioni per V_{M-} e V_{M+}.

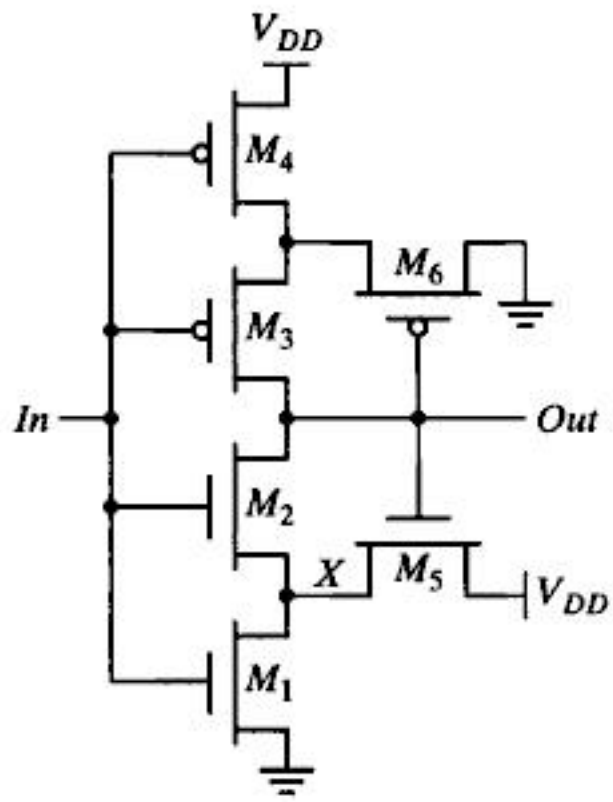


Figura 7-49 Trigger di Schmitt CMOS alternativo.

7.6.2 I circuiti monostabili

Un elemento monostabile è un circuito che genera un impulso di durata predeterminata ogni volta che è sollecitato da un impulso o da un evento di commutazione opportuno. Un circuito di questo tipo è detto *monostabile* perché possiede un solo stato stabile (quello a riposo).

Materiale protetto da copyright

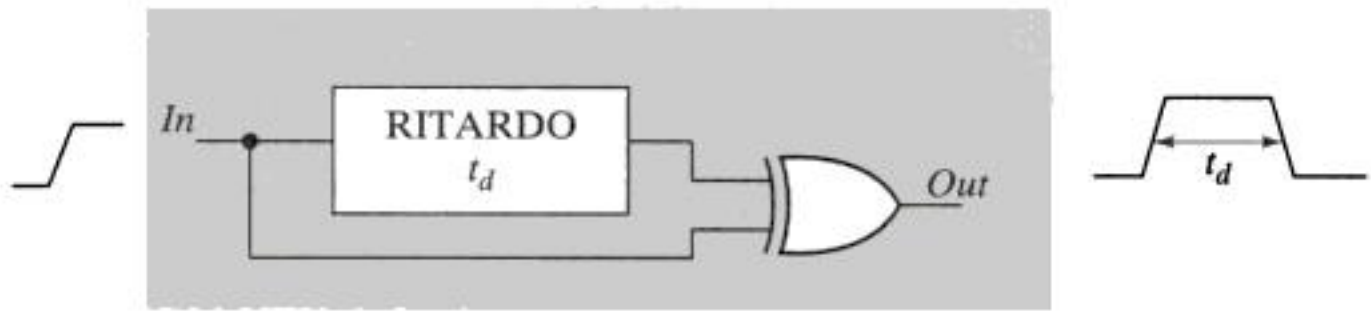


Figura 7-50 Circuito a colpo singolo innescato da transizione.

L'evento di innesco, che può essere la transizione di un segnale o un impulso, forza temporaneamente il circuito in un altro stato quasi stabile. Questo significa che il circuito tornerà necessariamente al suo stato originale di riposo, dopo un periodo di tempo determinato dai parametri del circuito stesso. Questo tipo di circuiti, noti anche come *a impulso singolo* (*one-shot*), è utile per generare impulsi di durata nota, funzionalità richiesta in molte applicazioni. Abbiamo già visto l'impiego di un circuito a impulso singolo per costruire i registri impulsivi. Un altro esempio ben noto è il circuito per il *rilevatore di transizioni di indirizzo* (ATD, *address transition detection*), utilizzato per la generazione del sincronismo nelle memorie statiche. Questo circuito si accorge di un cambiamento in un segnale o in un gruppo di segnali, per esempio quelli del bus indirizzi o dati, e genera un impulso per inizializzare la circuiteria seguente.

L'approccio più comune per l'implementazione di un circuito monostabile è l'uso di un semplice elemento di ritardo per controllare la durata dell'impulso generato. Questo concetto è illustrato nella Figura 7-50: nello stato quiescente entrambi gli ingressi alla XOR sono identici e l'uscita è bassa; una transizione all'ingresso fa sì che gli ingressi alla XOR siano temporaneamente diversi e l'uscita si porti al valore logico alto. Dopo un tempo t_d (il tempo di propagazione dell'elemento di ritardo), tale disturbo è esaurito e l'uscita ritorna a livello basso: è stato generato un impulso di durata t_d. L'elemento di ritardo può essere realizzato in molti modi, usando reti *RC* oppure con la cascata di porte logiche elementari.

7.6.3 I circuiti astabili

Un circuito astabile non possiede stati stabili. L'uscita oscilla avanti e indietro tra due stati quasi stabili, con un periodo determinato dalla topologia e dai parametri del circuito (ritardo, tensione di alimentazione, ecc.). Una delle applicazioni principali degli oscillatori è la generazione dei segnali di clock all'interno del chip (questa applicazione sarà discussa in dettaglio in uno dei prossimi capitoli, dedicati alla temporizzazione).

L'oscillatore ad anello (*ring oscillator*) è un esempio semplice di circuito astabile. È composto da un numero dispari di invertitori, connessi in una catena circolare. A causa del numero dispari di inversioni, non esiste alcun punto operativo stabile e il circuito oscilla con un periodo pari $2 \times t_p \times N$, dove N è il numero di invertitori nella catena e t_p è il tempo di propagazione di ciascun invertitore.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

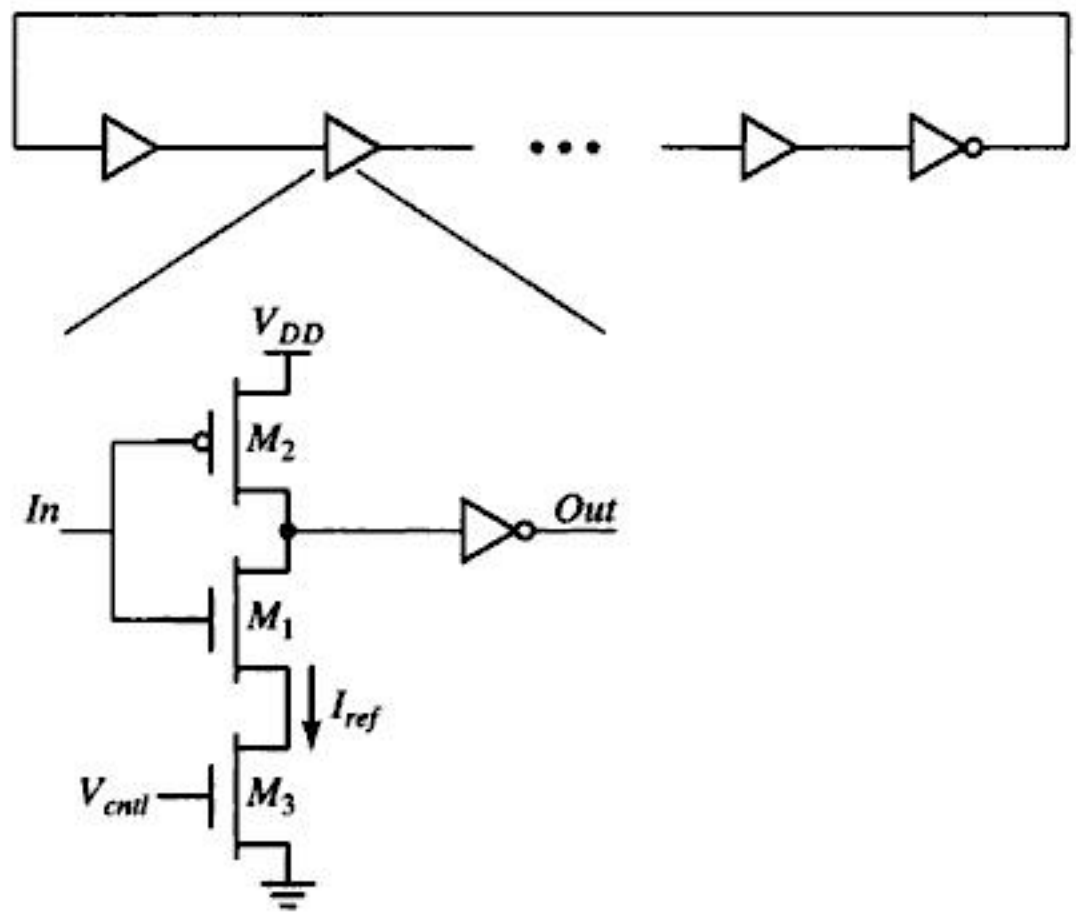


Figura 7-52 Oscillatore controllato in tensione realizzato con invertitori controllati in corrente.

l'invertitore controllato ha in uscita tempi di discesa lenti, dando luogo a correnti di corto circuito elevate. Una soluzione possibile è quella di mandare tale segnale in ingresso a un invertitore CMOS o, ancora meglio, a un trigger di Schmitt; è poi necessario un altro invertitore alla fine della catena, per fare in modo che il circuito oscilli.

Esempio 7.8 Simulazione dell'invertitore controllato in corrente

La Figura 7-53 mostra il ritardo simulato in un invertitore controllato in corrente in funzione della tensione di controllo V_c. Il ritardo dell'invertitore può variare su un intervallo ampio. Quando la tensione di controllo è minore della tensione di soglia, il dispositivo entra in regime di sottosoglia: questo si traduce in grosse variazioni del

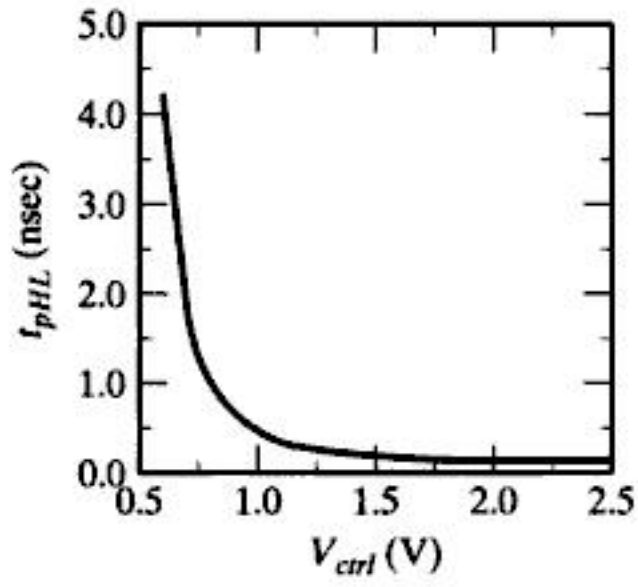
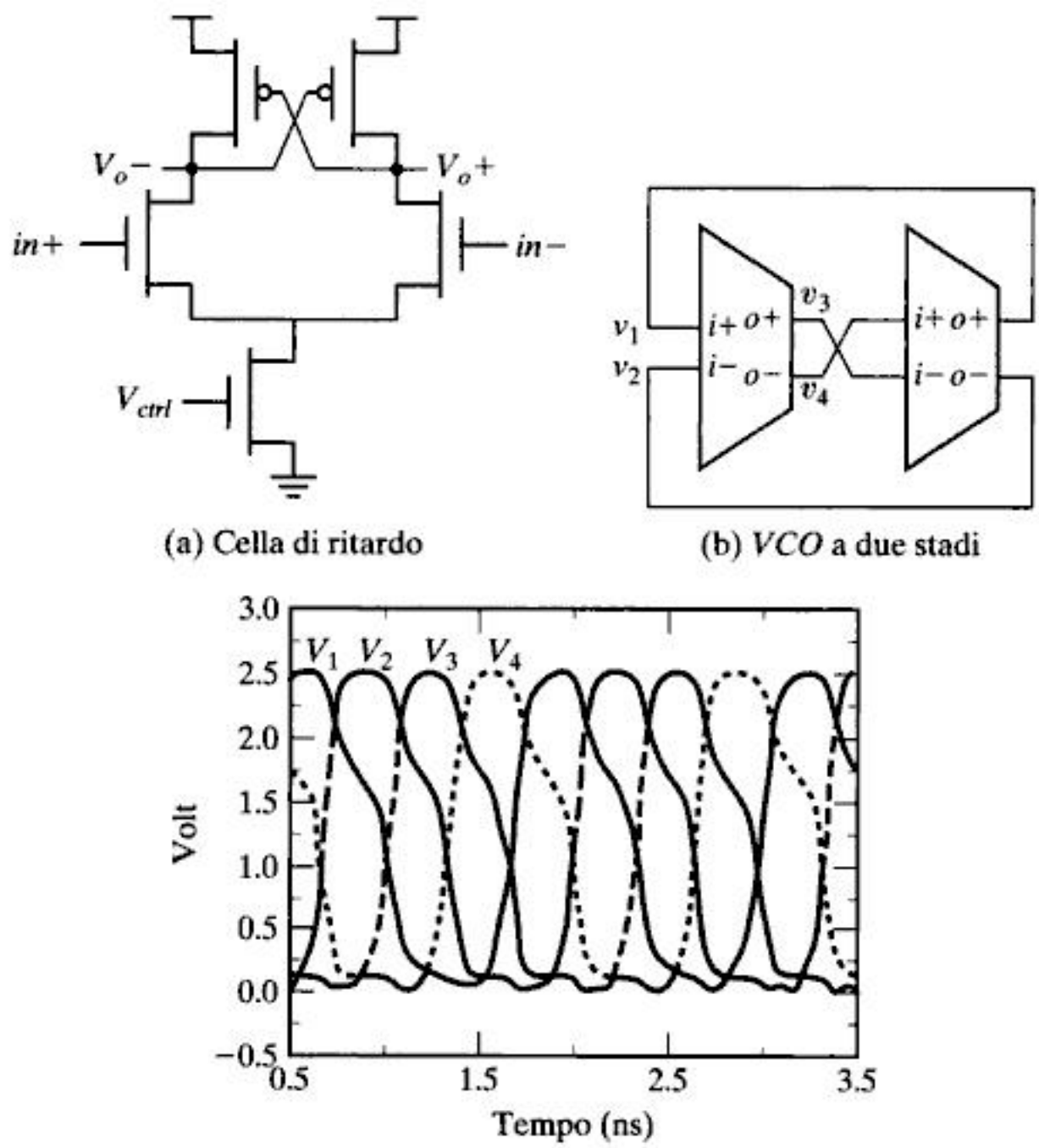


Figura 7-53 t_{pHL} in un invertitore controllato in corrente, in funzione della tensione di controllo.

Materiale protetto da copyright

ritardo, poiché la corrente di scarica diviene funzione esponenziale della tensione di controllo. Quando si opera in tale regione, il ritardo è molto sensibile alle variazioni della tensione di controllo e quindi anche al rumore.

Un altro approccio per realizzare la cella di ritardo consiste nell'uso di un elemento differenziale, come mostrato nella Figura 7-54a. Dato che la cella fornisce uscite sia invertenti che non invertenti, si può realizzare anche un oscillatore con un numero pari di elementi. La Figura 7-54b mostra un *VCO* differenziale a due stadi, in cui l'anello di retroazione fornisce uno sfasamento di 180° attraverso le due porte (una non invertente e una invertente), innescando l'oscillazione. Le forme d'onda simulate per questo *VCO* a due stadi sono mostrate nella Figura 7-54c. Le uscite in fase e in quadratura sono disponibili contemporaneamente. Un *VCO* di tipo differenziale ha anche maggiore immunità ai rumori di modo comune (per esempio rumore sull'alimentazione) rispetto al tipico oscillatore ad anello, ma consuma più potenza a causa della maggiore complessità e della corrente statica.



(a) Cella di ritardo

(b) *VCO* a due stadi

(c) Forme d'onda simulate per il *VCO* a due stadi

Figura 7-54 Elemento di ritardo differenziale e topologia del *VCO*.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

[Gonçalves83] N. Gonçalves and H. De Man, "NORA: a racefree dynamic CMOS technique for pipelined logic structures," *IEEE Journal of Solid-State Circuits*, vol. SC-18, no. 3, June 1983, pp. 261–266.

[Hill74] F. Hill and G. Peterson, *Introduction to Switching Theory and Logical Design*, Wiley, 1974.

[Jeong87] D. Jeong et al., "Design of PLL-based clock generation circuits," *IEEE Journal of Solid-State Circuits*, vol. SC-22, no. 2, April 1987, pp. 255–261.

[Kozu96] S. Kozu et al., "A 100 MHz 0.4 W RISC Processor with 200 MHz Multiply-Adder, using Pulse-register Technique," *IEEE ISSCC*, pp. 140–141, February 1996.

[Mano82] M. Mano, *Computer System Architecture*, Prentice-Hall, 1982.

[Montanaro96] J. Montanaro et al., "A 160-MHz, 32-b, 0.5-W CMOS RISC Microprocessor," *IEEE Journal of Solid-State Circuits*, pp. 1703–1714, November 1996.

[Mutoh95] S. Mutoh et al., "1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS," *IEEE Journal of Solid State Circuits*, pp. 847–854, August 1995.

[Partovi96] H. Partovi, "Flow-Through Latch and *Edge-Triggered* Flip-Flop Hybrid Elements," *IEEE ISSCC*, pp. 138–139, February 1996.

[Partovi01] H. Partovi, "Clocked Storage Elements," in *Design of High-Performance Microprocessor Circuits*, Chandakasan et al., ed., Chapter 11, pp. 207–233, 2001.

[Schmitt38] O. H. Schmitt, "A Thermionic Trigger," *Journal of Scientific Instruments*, vol. 15, January 1938, pp. 24–26.

[Suzuki73] Y. Suzuki, K. Odagawa, and T. Abe, "Clocked CMOS calculator circuitry," *IEEE Journal of Solid State Circuits*, vol. SC-8, December 1973, pp. 462–469.

[Yuan89] J. Yuan and Svensson C., "High-Speed CMOS Circuit Technique," *IEEE JSSC*, vol. 24, no. 1, February 1989, pp. 62–70.

Materiale protetto da copyright

PARTE

3

Il Sistema

"L'arte, mi sembra, dovrebbe semplificare. In questo sta quasi tutto il processo artistico più elevato: trovare di quali forme convenzionali e di quali dettagli si può fare a meno pur preservando lo spirito del tutto"

Willa Sibert Cather,
in "On the Art of Fiction" (1920).

"Semplicità e armonia sono le qualità che misurano il reale valore di ogni opera d'arte"

Frank Lloyd Wright.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

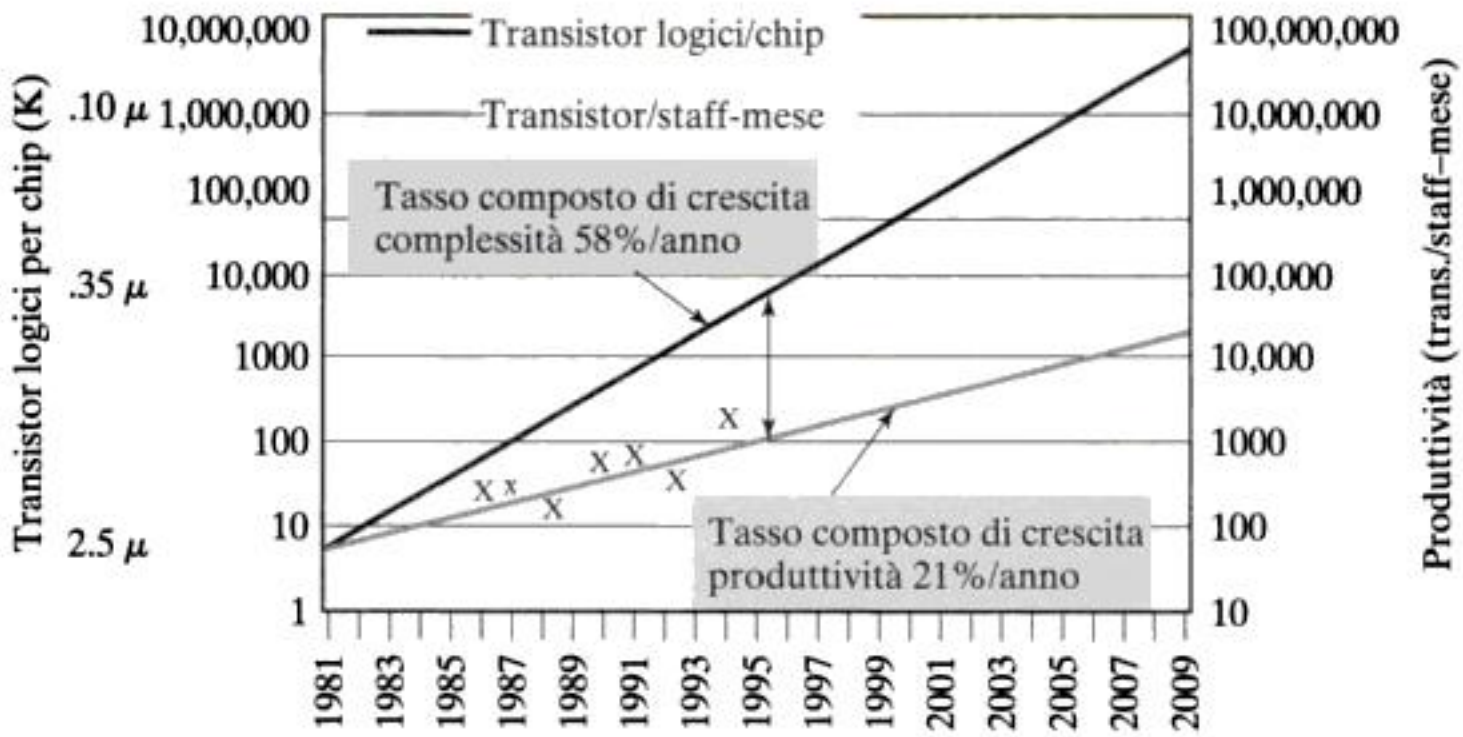


Figura 8-1 Il divario progettuale. Il passo di avanzamento della tecnologia (misurato in transistor logici per chip) supera la produttività (in transistor progettati in un mese da un singolo progettista). Da SIA [SIA97].

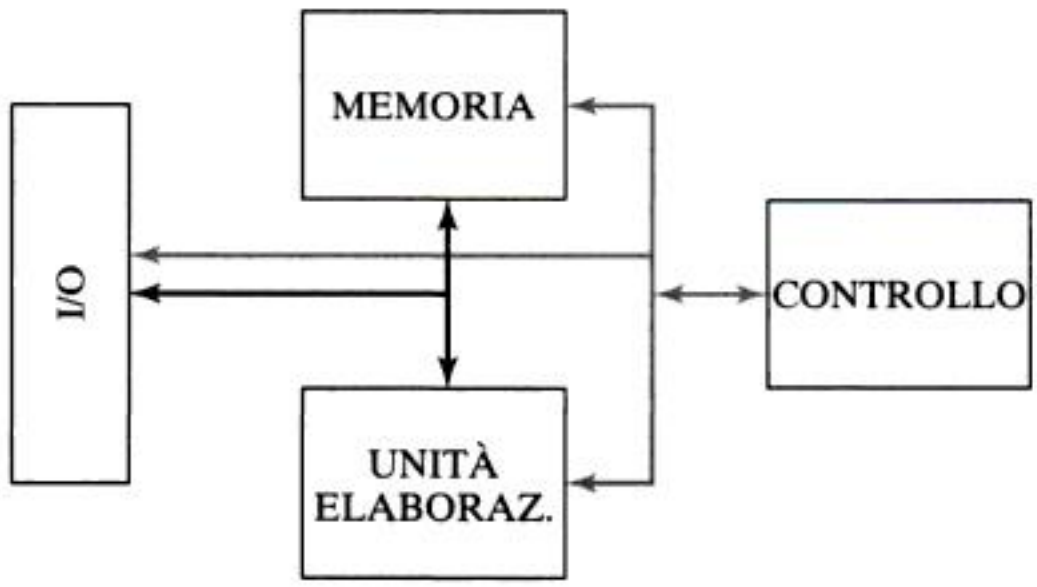


Figura 8-2 Composizione di un processore digitale generico. Le frecce rappresentano le interconnessioni possibili.

piuttosto che circuiti con interconnessioni flessibili e programmabili. La scelta di una particolare piattaforma implementativa è per lo più influenzata dal compromesso ottimo tra varie figure di merito del sistema, quali area, velocità, potenza, tempo di progetto e grado di riusabilità.

- **Il modulo di controllo** determina quale particolare azione accade nel processore per un dato istante temporale. Un controllore può essere visto come una macchina a stati finiti; è formato da registri e logica e quindi è un circuito sequenziale. La logica può essere implementata in modi diversi: o come interconnessione di porte logiche fondamentali (usando standard cell) o in modo più strutturato usando PLA e memorie con la sequenza di istruzioni.
- **Il modulo di memoria** svolge il ruolo di area centralizzata per l'immagazzinamento dei dati. Esiste una vasta gamma di classi di memorie differenti. La differenza principale tra queste classi risiede nel modo in cui è possibile accedere ai dati, come "a sola lettura" piuttosto che "a lettura e scrittura", ad accesso sequenziale piuttosto che

Materiale protetto da copyright

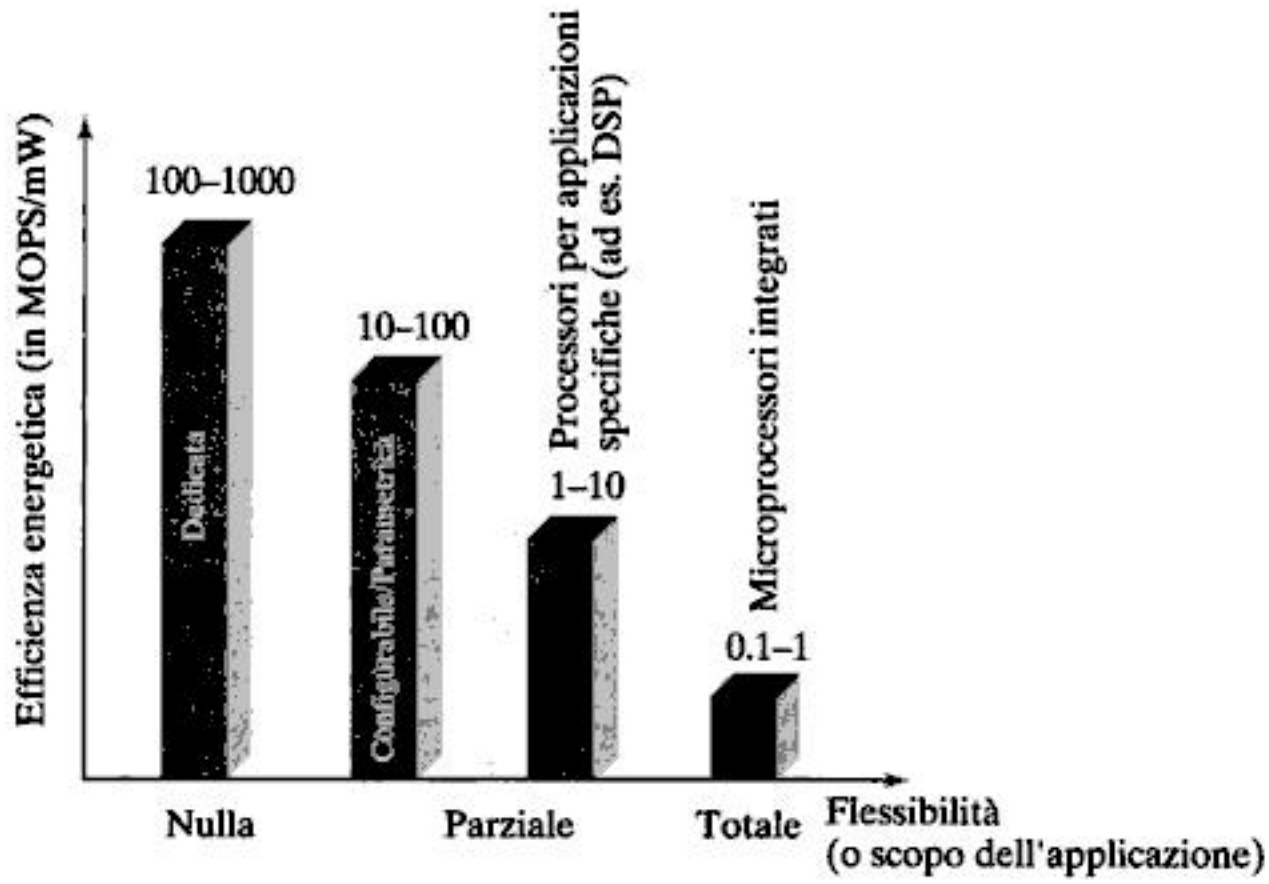


Figura 8-4 Il compromesso tra flessibilità ed efficienza energetica, misurata in MOPS/mW (milioni di operazioni per mJ di energia) per differenti tecnologie realizzative. I dati sono stati raccolti per un processo CMOS da 0.25 μm [Rabaey00].

può osservare un'impressionante differenza di **tre ordini di grandezza**. Questo dimostra chiaramente che ogniqualvolta l'efficienza energetica è una priorità, sono da preferire implementazioni hard-wired o comunque stili con un grado di flessibilità limitato (quali moduli configurabili o parametrici).

In questo capitolo e nei tre successivi, prendiamo in esame le tecniche implementative per la logica generica e i controllori (questo capitolo), per le interconnessioni (Capitolo 9), per le unità di elaborazione (Capitolo 11) e per le memorie (Capitolo 12). Teniamo sempre presente che la scelta dell'approccio implementativo può avere un effetto significativo sulla qualità del prodotto finale. Per la progettazione di sistemi complessi che includono molti blocchi e che quindi necessitano di un'attenzione speciale, divengono aspetti importanti la sincronizzazione e temporizzazione (Capitolo 10) e le reti di distribuzione delle alimentazioni (Capitolo 9). La distribuzione dei segnali di clock e delle alimentazioni è per esempio divenuta uno dei problemi dominanti nella progettazione di processori allo stato dell'arte. Un certo numero di inserti dedicati alle metodologie progettuali, distribuiti tra i vari capitoli, affrontano le peculiarità tipiche della progettazione di questi componenti complessi e introducono gli strumenti di sintesi avanzati e automatici che sono a disposizione del progettista. Gli inserti F, G e H discutono rispettivamente la sintesi circuitale, la verifica e la misura sperimentale.

Materiale protetto da copyright

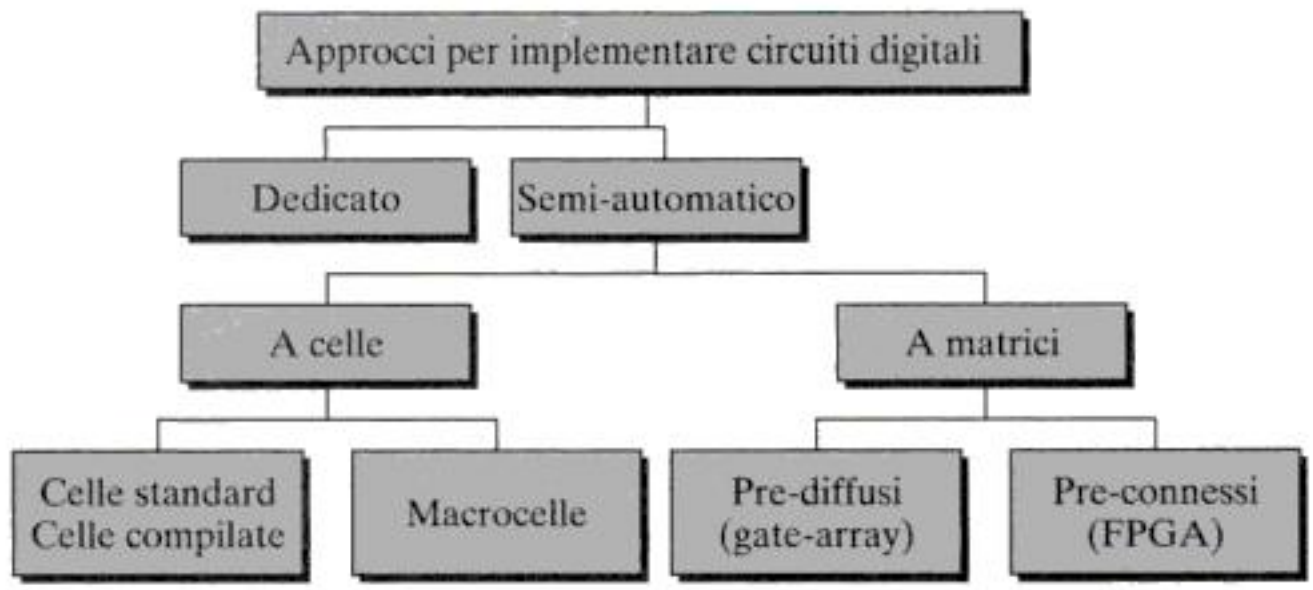


Figura 8-5 Panoramica delle tecniche realizzative per circuiti integrati digitali (secondo [DeMicheli94]).

8.2 Dall'approccio dedicato a quello semi-automatico a celle standard

La fattibilità di un sistema microelettronico dipende da una serie di fattori (spesso) in conflitto, quali le prestazioni in termini di velocità e consumo di potenza e i volumi di produzione. Per esempio, per essere competitivo sul mercato, un microprocessore deve eccellere in prestazioni, ma avere un prezzo basso per il cliente. Ottenere entrambi questi risultati simultaneamente è possibile solamente se il prodotto è venduto su larga scala: i costi elevati associati alla progettazione di sistemi ad alte prestazioni devono essere ammortizzati su molti componenti. Applicazioni quali supercomputer e alcuni sistemi per la difesa prospettano uno scenario differente: avendo la massima prestazione come obiettivo primario, le tecniche progettuali completamente dedicate (*full-custom*) sono spesso le più adatte. Il volume di produzione è limitato, ma il costo delle componenti elettroniche è solo una frazione del costo totale del sistema e quindi non rappresenta un problema rilevante. Infine l'obiettivo primario in molte applicazioni dell'elettronica di consumo non è l'aumento delle prestazioni, ma quello del livello di integrazione. In queste condizioni, il costo della progettazione può essere ridotto sostanzialmente, usando le tecniche avanzate di progettazione automatica, che accettano alcuni compromessi in termini di prestazioni, ma minimizzano il tempo di progetto. Come sottolineato nel Capitolo 1, il costo di un dispositivo a semiconduttore è la somma di due componenti:

- I *costi fissi*, che si affrontano una sola volta per ogni sistema e comprendono i costi per progettare il componente.
- Il *costo di produzione per componente*, che è funzione della complessità del processo, dell'area del sistema e della resa del processo

Queste considerazioni di carattere economico hanno spronato lo sviluppo di tecniche realizzative diverse che spaziano da sistemi fatti *a mano* ad altissime prestazioni a sistemi standard completamente programmabili di medie o basse prestazioni. La Figura 8-5 fornisce una panoramica delle diverse metodologie. Nei paragrafi che seguono, descriviamo dapprima la progettazione completamente dedicata, quindi gli approcci semi automatici e basati su matrici logiche.

Materiale protetto da copyright

8.3 La progettazione dedicata

Quando le prestazioni o la densità di integrazione sono di primaria importanza, progettare a mano il sistema dalla topologia circuitale fino a livello fisico sembra essere l'unica possibilità. In effetti questo approccio era l'unico disponibile all'inizio dell'era dei circuiti microelettronici, com'è adeguatamente dimostrato dal microprocessore Intel 4004 (vedi Figura 1-5a). Il notevole impegno richiesto dalla progettazione dedicata si traduce in costi e tempo di immissione sul mercato (*time to market*) elevati. Questo approccio è quindi giustificabile dal punto di vista economico solo nelle condizioni seguenti:

- Il blocco dedicato può essere riusato molte volte (per esempio, nel caso di una cella di libreria).
- Il costo può essere ammortizzato su volumi molto elevati. Microprocessori e memorie a semiconduttore sono esempi di applicazioni che appartengono a questa classe.
- Il costo non è un criterio primario, come nel caso di supercomputer o ipersupercomputer.

Con i progressi continui nel campo dell'automazione della progettazione, la percentuale di sistemi progettati in modo dedicato si riduce di anno in anno. Persino nei microprocessori più avanzati ad alte prestazioni, quali il processore Intel Pentium® 4 (vedi Figura 8-6), praticamente ogni sezione è stata progettata automaticamente con tecniche semi-dedicate. Solamente i moduli più critici dal punto di vista delle prestazioni, quali gli anelli ad aggancio di fase (PLL, *phase locked loop*) e i buffer dei clock sono stati progettati manualmente. Di fatto la progettazione di celle di libreria è l'unica area dove ancora oggi la progettazione dedicata prospera.

Il livello di automazione del flusso di progetto dedicato è minimo, sebbene alcuni strumenti si sono dimostrati indispensabili. In sintonia con una vasta gamma di strumenti per la verifica, simulazione, estrazione e modellazione, gli editor di layout e gli strumenti di verifica delle regole di layout e di quelle elettriche (come già descritto nell'Inserto A) sono il nucleo di qualsiasi ambiente di progettazione dedicata. Un'ottima trattazione delle potenzialità e delle difficoltà della progettazione dedicata si può trovare in [Grundman97].

8.4 Metodologie progettuali basate su celle

Dato che l'approccio dedicato si è dimostrato oltremodo costoso, negli anni è stata introdotta una vasta gamma di metodologie progettuali per abbreviare e automatizzare il flusso di progetto. Tale automazione si paga in termini di riduzione della densità di integrazione e delle prestazioni. In generale possiamo affermare che: **più breve è il tempo di progettazione, maggiore è la penalità da accettare**. In questo paragrafo, prendiamo in esame un certo numero di approcci progettuali che comunque richiedono di ripercorrere interamente tutti i passi del processo per ogni nuovo progetto. Gli approcci progettuali *basati su matrici logiche*, discussi nei paragrafi seguenti, riducono ulteriormente il tempo di progetto e i costi, coinvolgendo solo una parte dei passi di processo o essendo addirittura indipendenti dal processo di fabbricazione.

Materiale protetto da copyright

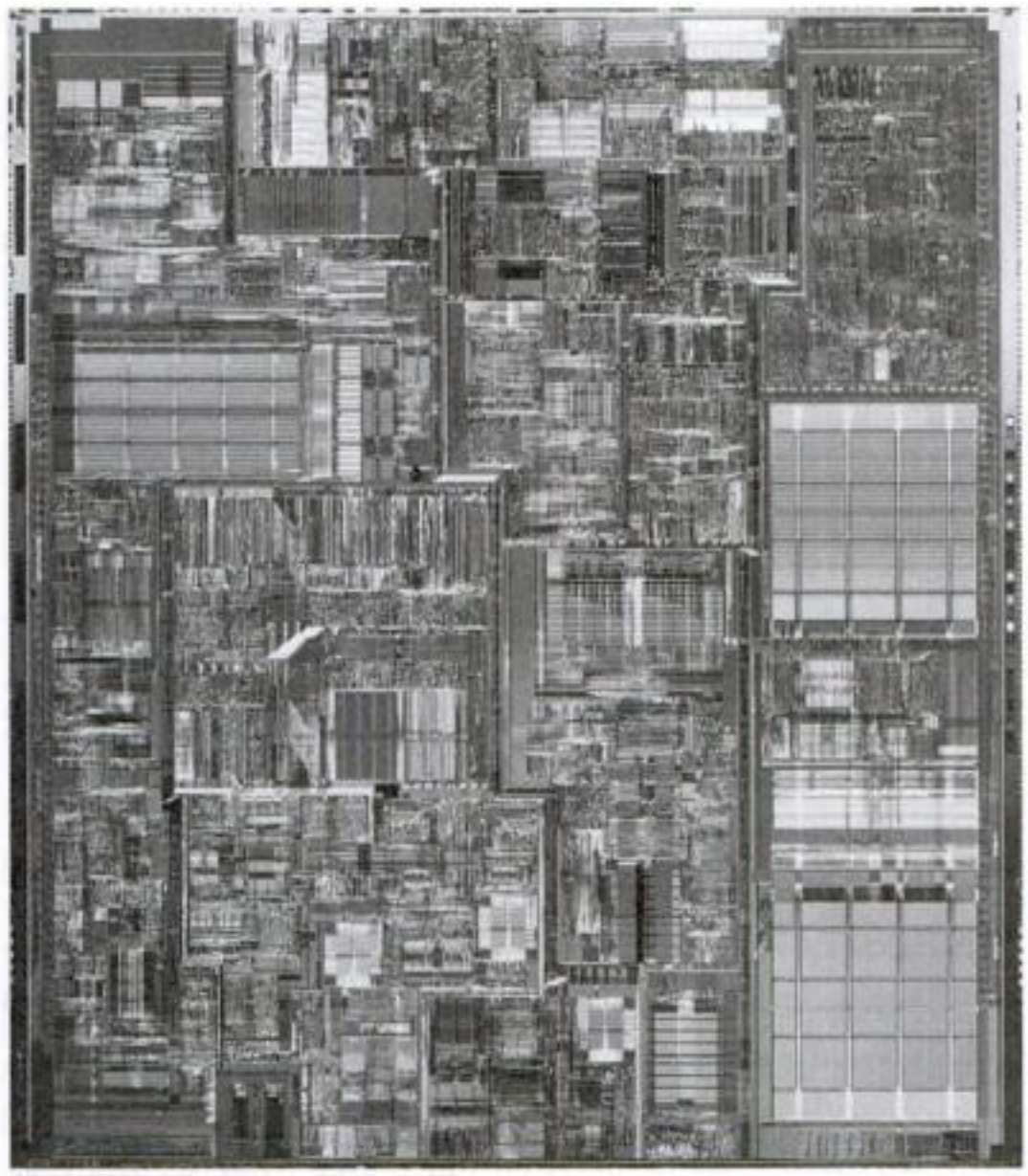


Figura 8-6 Fotografia del chip del processore Intel Pentium® 4. Contiene 42 milioni di transistor, progettati in una tecnologia CMOS da 0.18 μm. La prima generazione lavora a una velocità di clock di 1.5 GHz (gentile concessione di Intel Corp).

L'idea alla base della progettazione a celle è di ridurre lo sforzo per l'implementazione *riusando* una libreria di celle limitata. Il vantaggio di questo approccio è che le celle devono essere progettate e verificate una sola volta per una data tecnologia e possono poi essere riusate molte volte, ammortizzando quindi il costo della loro fase di progetto. Lo svantaggio risiede nell'essere vincolati a una libreria che riduce la possibilità di raffinare il circuito. Gli approcci a celle si dividono in diverse classi, secondo la granularità degli elementi della libreria.

8.4.1 Celle standard

L'approccio basato su celle standard rende la porta logica l'unità elementare del flusso di progetto. Una libreria solitamente contiene una vasta gamma di porte logiche, in varie versioni per coprire un intervallo ampio di fan-in e fan-out. Oltre alla funzioni logiche fondamentali, quali invertitori, AND/NAND, OR/NOR, XOR/XNOR e flip-flop, una libreria solitamente include anche funzioni più complesse, quali AND-OR-INVERT, multiplexer, full adder, comparatori, contatori, e codificatori. La fase di sintesi parte quindi da uno schema circuitale che contiene solo blocchi disponibili nella libreria, oppure tale rappresentazione è generata automaticamente partendo da una descrizione a livello più alto. Da qui il layout del circuito è generato automaticamente: è infatti possibile sfruttare un livello di auto-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

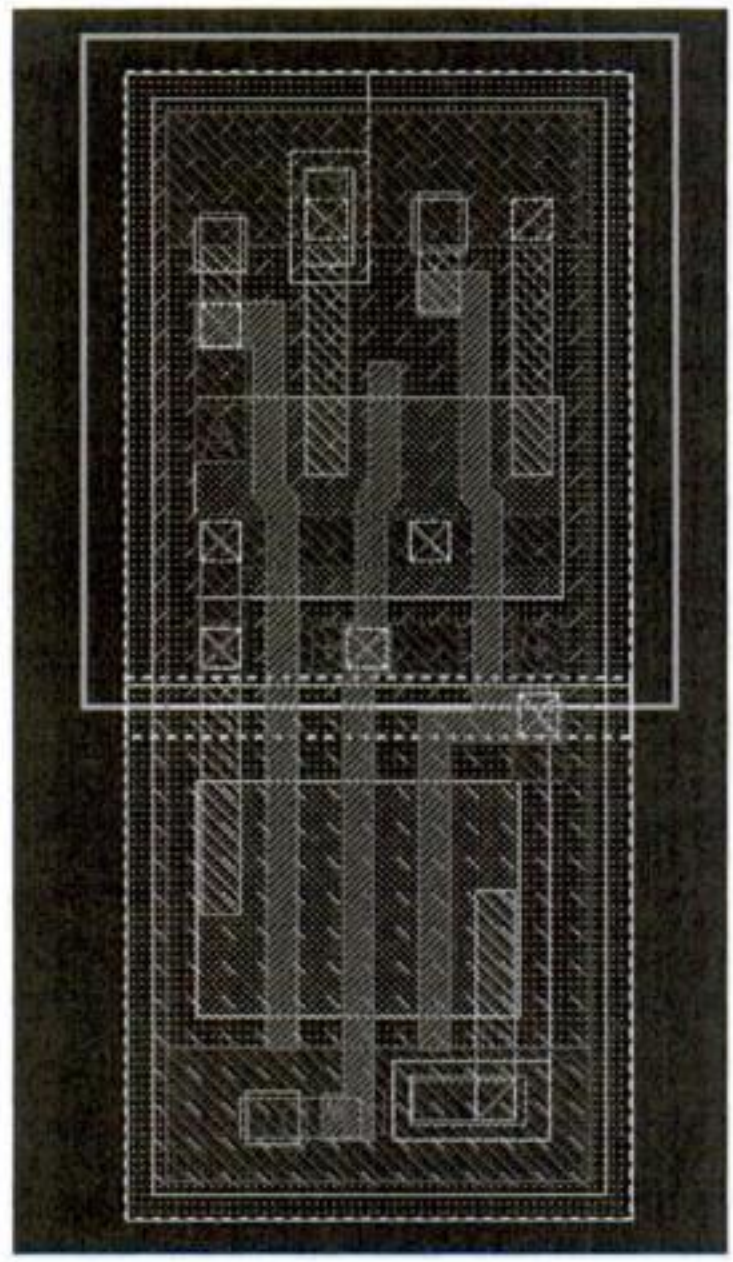


Figura 8-9 NAND a tre ingressi a celle standard (Gentile concessione ST Microelectronics).

vedendo capacità di carico da 0.18 pF fino a 0.72 pF e occupando un'area da 16.4 μm^2 fino a 32.8 μm^2. La cella mostrata è la versione a bassa prestazione ma ad alta efficienza energetica e usa transistor con tensione di soglia elevata per ridurre le correnti di perdita. I transistor NMOS e PMOS nelle reti di pull-down e pull-up hanno un fattore di forma (W/L) di circa 8.

Osserviamo come la strategia del layout segua le linee guida mostrate nella Figura D-2. Le linee di alimentazione sono distribuite orizzontalmente e condivise dalle celle nella stessa riga. I segnali di ingresso sono instradati verticalmente usando piste di polisilicio. I terminali di ingresso e uscita sono posizionati all'interno del corpo della cella (come evidenziato dai simboli dei *pin* nel disegno del layout), in accordo all'approccio di stendere le connessioni al di sopra delle celle delle metodologie a celle standard contemporanee.

L'approccio a celle standard è divenuto estremamente comune ed è usato per la realizzazione della quasi totalità degli elementi logici nei circuiti integrati contemporanei. L'unica eccezione è rappresentata dai quei casi in cui si richiedono prestazioni molto elevate e consumi molto ridotti oppure quando la struttura delle funzioni da realizzare è molto regolare

Materiale protetto da copyright

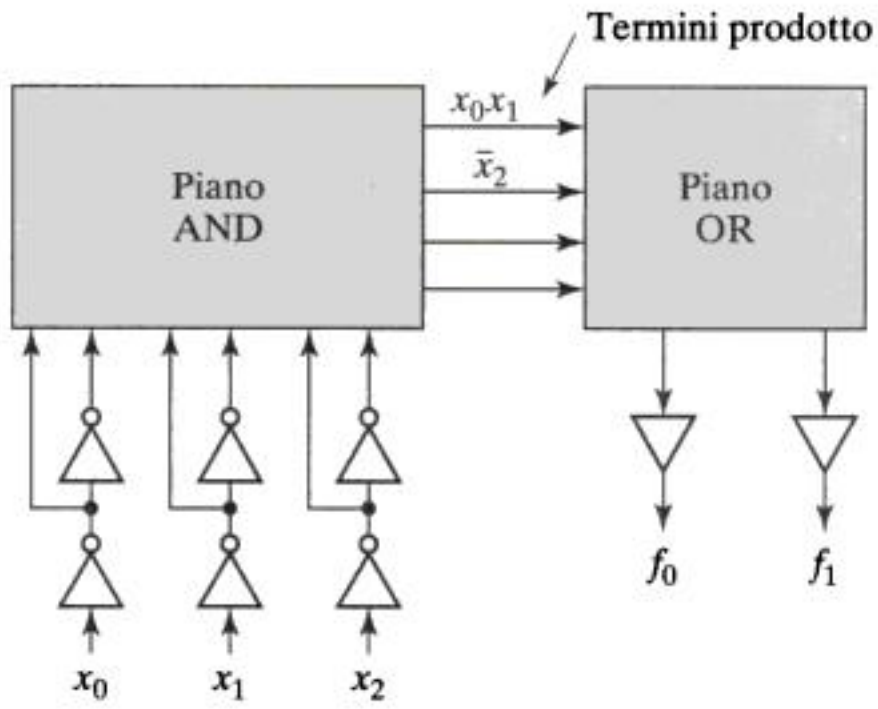


Figura 8-10 Implementazione regolare a due livelli di funzioni booleane.

Il concetto può essere meglio compreso con l'aiuto di un esempio. Consideriamo la seguente funzione logica, per la quale abbiamo trasformato le equazioni in forma di somma di prodotti usando l'algebra booleana:

$$\begin{aligned} f_0 &= x_0x_1 + \overline{x_2} \\ f_1 &= x_ox_1x_2 + \overline{x_2} + \overline{x_0}x_1 \end{aligned} \tag{8.1}$$

Un vantaggio importante di questo tipo di rappresentazione è che se ne può concepire facilmente una realizzazione regolare, come mostrato nella Figura 8-10. Un primo livello di porte logiche realizza le operazioni AND (chiamate anche *termini prodotto* o *mintermini*) mentre un secondo livello realizza le funzioni OR, dette *termine somma.* Quindi un PLA è una macrocella rettangolare, costituita da una matrice di transistor allineati in righe in corrispondenza dei termini prodotto e in colonne in corrispondenza di ingressi e uscite. Le colonne di ingresso e uscita dividono la matrice in due sotto-matrici, chiamate rispettivamente piano AND e piano OR.

Lo schema circuitale della Figura 8-10 non può essere realizzato direttamente poiché le funzioni logiche a singolo stadio CMOS sono sempre invertenti. Con alcuni semplici passaggi di algebra booleana l'Eq. (8.1) può però essere riscritta in forma NOR–NOR:

$$\begin{aligned} \overline{f_0} &= \overline{\overline{(\overline{x_0} + \overline{x_1})} + \overline{x_2}} \\ \overline{f_1} &= \overline{\overline{(\overline{x_0} + \overline{x_1} + \overline{x_2})} + \overline{x_2} + \overline{(x_0 + \overline{x_1})}} \end{aligned} \tag{8.2}$$

Problema 8.1 Rappresentazioni logiche a due livelli

È ugualmente possibile rappresentare l'Eq. (8.1) in forma NAND–NAND. In linea generale, la rappresentazione NOR–NOR è preferibile a causa dell'inaccettabile lentezza di porte NAND con fan-in elevati. La configurazione NAND–NAND è però molto densa e questo può essere di aiuto per ridurre la dissipazione di potenza. Ricavare la rappresentazione NAND–NAND per l'esempio dell'Eq. (8.2).

Materiale protetto da copyright

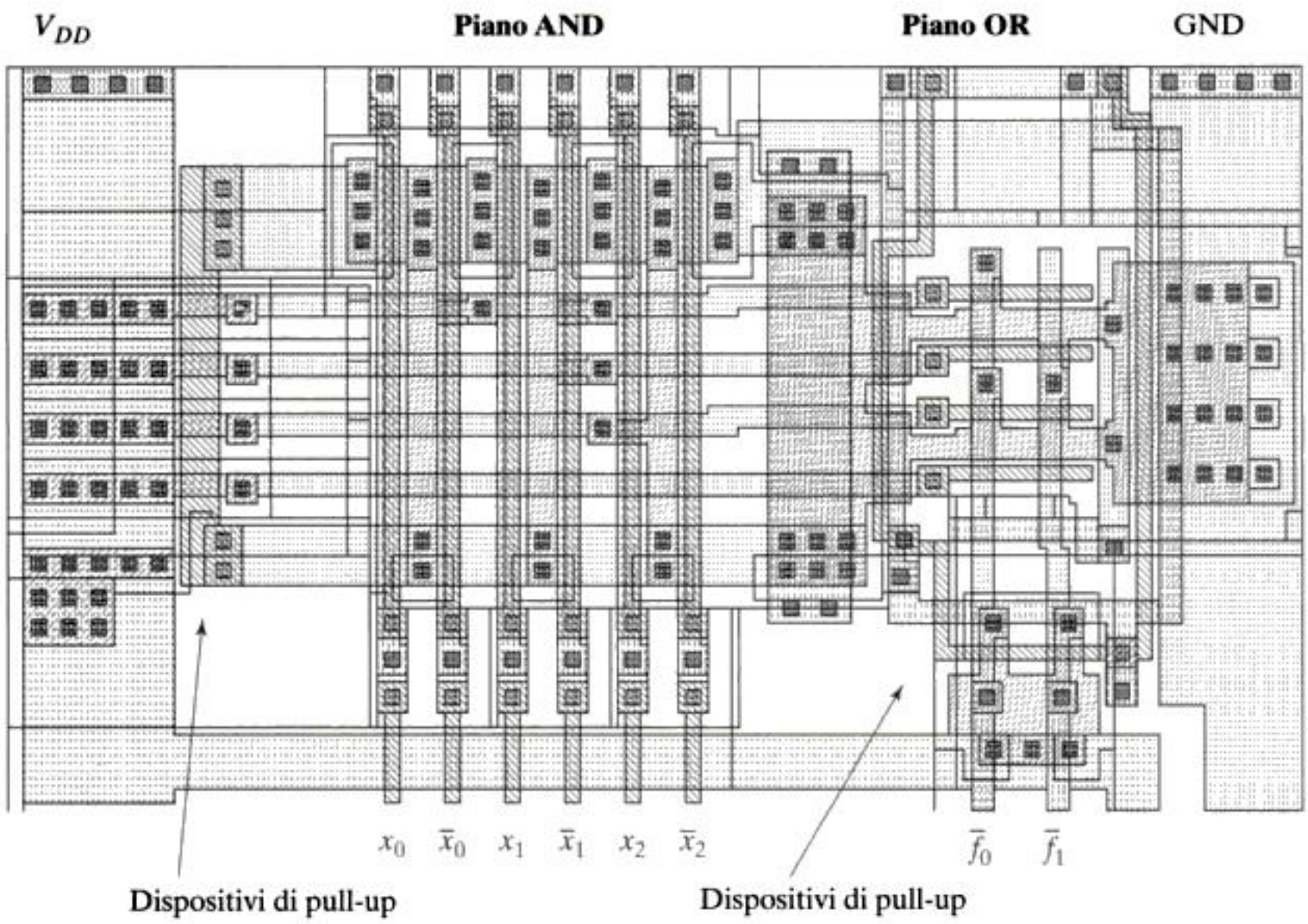


Figura 8-11 Layout di una PLA che implementa l'Eq. (8.2).

Tradurre un insieme di funzioni logiche a due livelli in un circuito fisico si riduce ora a un compito di "programmazione". Si tratta cioè di decidere cioè dove piazzare i transistor sia nel piano AND che in quello OR. Questo compito può essere facilmente automatizzato, da cui il rapido successo delle PLA. Una generazione automatica dell'implementazione tramite PLA della funzione logica dell'Eq. (8.2) è mostrata nella Figura 8-11. Sfortunatamente la struttura regolare, sebbene prevedibile, implica una ridondanza notevole in termini di area e ritardi (com'è piuttosto evidente nel layout). Questo è stato il fattore che ha segnato la fine definitiva delle PLA nel mondo della progettazione semi-automatica. La realizzazione a livello di transistor delle PLA sarà discussa nel Capitolo 12. ■

8.4.2 Celle compilate

Il costo dello sviluppo e della caratterizzazione di una libreria di celle non deve essere sottostimato. Le librerie moderne contengono da diverse centinaia fino a più di un migliaio di celle. Queste celle vanno poi riprogettate ogni volta che si migra a una nuova tecnologia. Inoltre anche nell'ambito di una data tecnologia si possono verificare delle modifiche del processo. Per esempio, per migliorare la resa, vengono spesso modificate la larghezza minima delle piste di metallo o le regole riguardanti i contatti; di conseguenza il disegno del layout e la caratterizzazione di una data libreria devono essere eseguiti nuovamente. Inoltre, persino una libreria molto ricca di componenti ha comunque lo svantaggio di essere limitata, di offrire cioè un numero finito di opzioni progettuali. Quando si ottimizzano prestazioni o consumi, diviene interessante la possibilità di avere celle dedicate con il dimensionamento dei transistor ottimizzato. Con l'aumento poi degli effetti di carico delle interconnessioni, è

Materiale protetto da copyright

irrinunciabile sia dal punto di vista delle prestazioni che dei consumi avere delle celle che permettano di modificare le dimensioni dei buffer [Sylvester98], da cui la ricerca per la generazione automatica (o compilata) delle celle.

Sono stati escogitati diversi approcci automatici per la generazione istantanea del layout, data la rete di transistor, ma il layout di celle automatico e di alta qualità è rimasto un obiettivo inafferrabile. I primi approcci si basavano su topologie predefinite; le soluzioni successive introdussero una maggiore flessibilità nel piazzamento dei transistor (per esempio, [Hill85]). Densità di layout prossime a quanto può essere raggiunto da un progettista umano sono ora a portata di mano ed è disponibile un certo numero di strumenti commerciali, per esempio [Cadabra01, Prolific01].

Esempio 8.3 Generazione automatica delle celle

Il flusso del tipico processo di generazione di celle è mostrato attraverso l'esempio di un semplice invertitore (usando lo strumento Abracad [Cadabra01]).

- Prima di tutto viene sviluppato lo schema circuitale della cella. La netlist di SPICE è il punto di partenza per la generazione automatica del layout. Il generatore esamina la netlist, cominciando dalle geometrie dei transistor. Nel caso di un invertitore CMOS, la cella contiene solamente due transistor (vedi Figura 8-12a).
- Lo strumento prosegue lungo lo stesso percorso che sarebbe seguito da un progettista. I transistor vengono piazzati all'interno di una cella che rispetta lacune

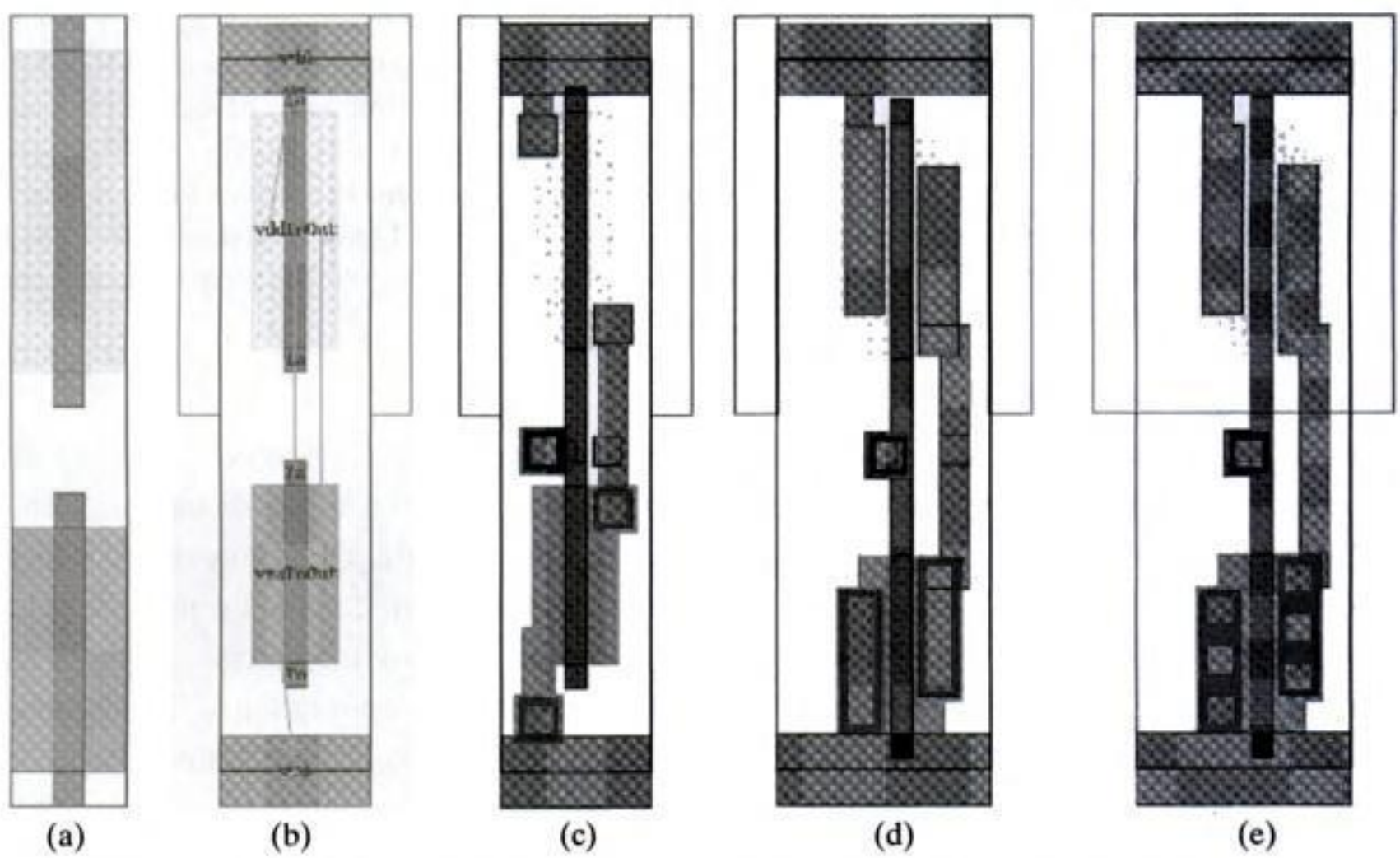


Figura 8-12 Celle di layout automatiche (a) geometria iniziale dei transistor, (b) transistor piazzati con l'indicazione delle interconnessioni previste, (c) cella iniziale con connessioni, (d) cella compatta, (e) cella finita.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

La disponibilità di macromoduli ha mutato sostanzialmente il panorama della progettazione semi-automatica nel ventunesimo secolo. Con la complessità dei circuiti integrati che cresce esponenzialmente, l'idea di riprogettare da capo ogni nuovo circuito integrato diviene un proposito antieconomico e non plausibile. I circuiti vengono costruiti sempre più sulla base del riuso di blocchi fondamentali di complessità e funzionalità crescenti. Tipicamente questi moduli vengono acquistati da terzi, che rendono disponibili le loro funzioni sulla base di accordi di diritti d'autore o licenze. I macromodelli distribuiti in questo modo sono detti moduli di *proprietà intellettuale* (IP, *intellectual property*). Questo approccio è in qualche modo comparabile al mondo del software, dove un grosso progetto di programmazione sfrutta intensamente il riuso di librerie di software. Buoni esempi di moduli IP abitualmente disponibili sono i microprocessori, i microcontrollori, i processori DSP, le interfaccie per i bus (come le PCI) e molte funzioni per scopi specifici, quali FFT e filtri per applicazioni DSP, codice di correzione di errore per comunicazioni wireless e codifica e decodifica MPEG per applicazioni video. Ovviamente, affinché un modulo IP sia utile, non deve limitarsi a mettere a disposizione l'hardware, ma deve anche essere distribuito con gli strumenti software del caso (quali compilatori e strumenti di verifica per i processori integrati), modelli di predizione e configurazioni di misura. Questi ultimi sono piuttosto importanti, perché sono l'unico mezzo a disposizione dell'utente finale per verificare che il modulo realizzi la funzionalità e le prestazioni promesse.

La progettazione di un intero sistema su singolo chip sta divenendo rapidamente un esercizio di riuso a diversi gradi di granularità. Al livello inferiore troviamo le librerie di celle standard; a livello superiore abbiamo i moduli funzionali quali moltiplicatori, unità di elaborazione e memorie; quindi troviamo i processori integrati e infine le megacelle per applicazioni specifiche. Poiché smpre più funzioni di un sistema confluiscono su un singolo chip, non sorprende vedere che un tipico ASIC consiste di una mistura di stili di progettazione e moduli, integrando una serie di macrocelle hard o soft con una marea di celle standard.

Esempio 8.6 Un processore per comunicazioni wireless

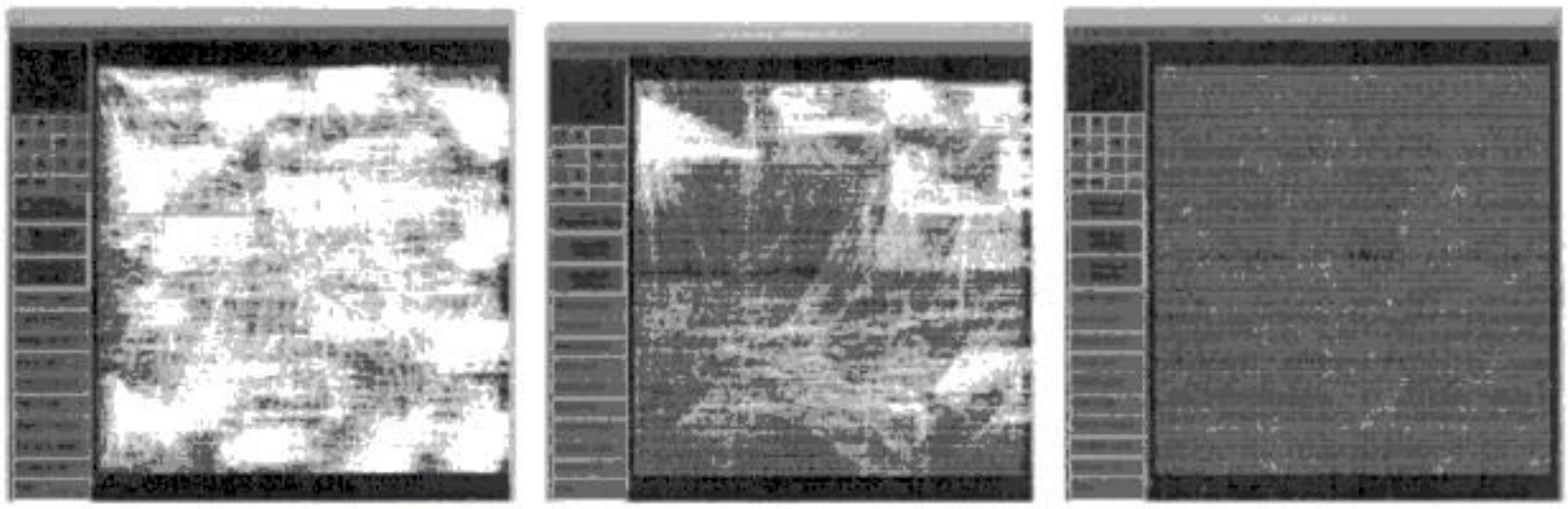
La Figura 8-15 mostra un circuito integrato che implementa i vari strati del protocollo per un sistema di comunicazioni wireless al coperto [Silva01]. La maggior parte dell'area è occupata dal microprocessore integrato (il processore Tensilica Xtensa [Xtensa01]) e dalla sua memoria di sistema. Questo processore realizza un'implementazione flessibile dei livelli più alti degli strati del protocollo (applicazione/rete), e permette cambiamenti nella funzionalità del chip anche dopo la fabbricazione. I moduli di memoria sono stati generati usando il compilatore di moduli messo a disposizione dal venditore del processo. Il processore è stato generato automaticamente a partire da una descrizione ad alto livello in Verilog e usa celle standard per la sua implementazione fisica. Il vantaggio di usare l'approccio "soft" sta nel fatto che l'insieme di istruzioni del processore possono essere adattate all'applicazione e che il processore stesso può essere facilmente esportato a tecnologie di fabbricazione diverse.

Implementare la parte di calcolo più pesante del protocollo (MAC/PHY) sul microprocessore richiederebbe una frequenza di clock molto elevata e incrementerebbe senza

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Sistema iniziale (b) Sistema intermedio (c) Sistema finale

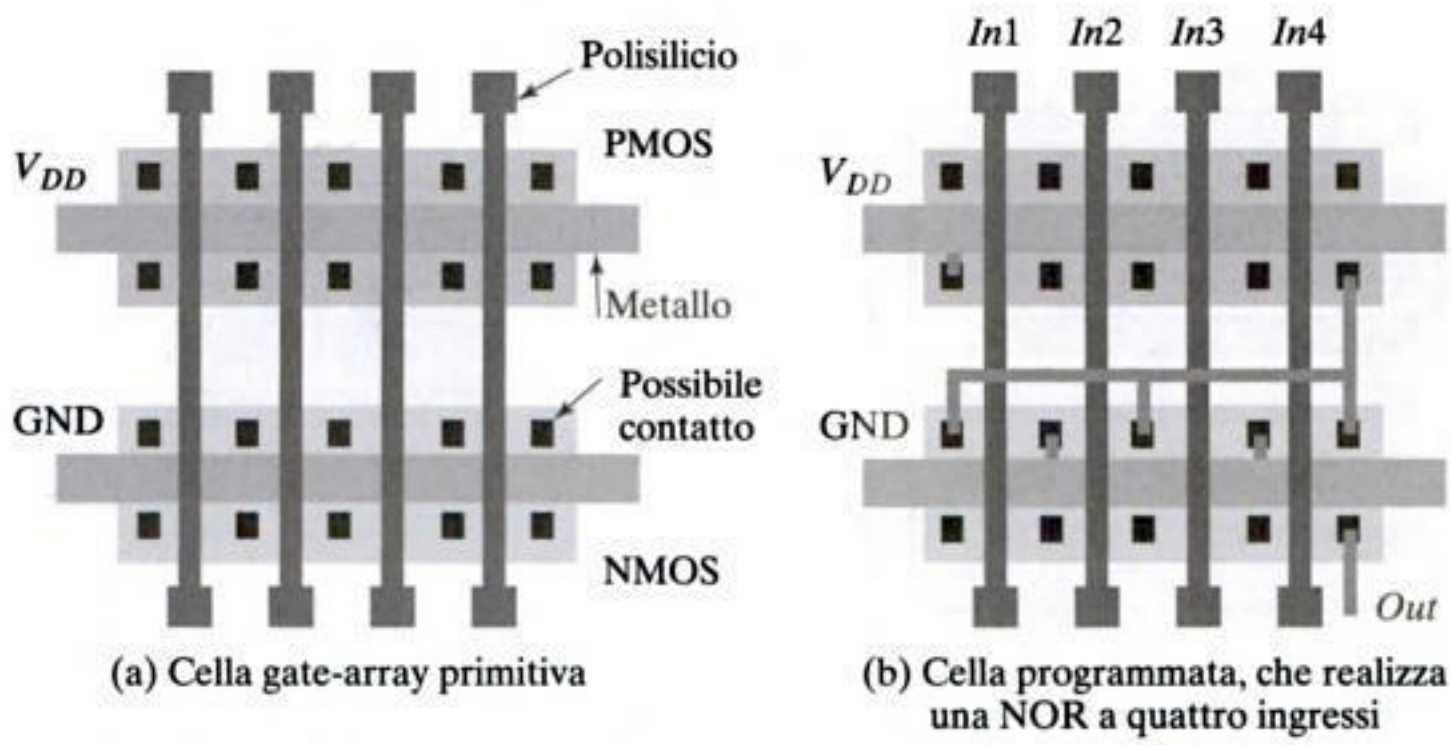
Figura 8-17 Processo di timing closure. Le linee bianche indicano reti con violazioni dei vincoli temporali. A ciascuna iterazione del processo di progettazione gli errori di temporizzazione vengono rimossi ottimizzando la logica, inserendo dei buffer, vincolando il piazzamento o ottimizzando la fase di connessione finché si ottiene un sistema privo di errori [Avanti01].

9. **Tape Out**. Una volta che il circuito soddisfa tutti gli obiettivi e tutte le funzioni, viene generato un file binario che contiene tutte le informazioni necessarie alla generazione delle maschere. Questo file è quindi inviato al venditore di circuiti integrati o alla fonderia. Questo momento fondamentale della vita di un chip è detto *tape out*.

Sebbene il flusso di progettazione appena descritto abbia soddisfatto le nostre esigenze per molti anni, ha evidenziato molte lacune da quando la tecnologia CMOS ha raggiunto il limite dei 0.25 μm. Con il procedere della tecnologia verso le dimensioni submicrometriche e decananometriche, i parassiti associati al layout (specialmente quelli dovuti alle interconnessioni) stanno assumendo un ruolo sempre più predominante. I modelli di previsione usati dagli strumenti di sintesi logica e strutturale hanno diverse difficoltà a fornire una stima accurata di questi parassiti. La probabilità che un circuito generato soddisfi i vari vincoli temporali al primo tentativo è quindi molto bassa (Figura 8-17a). Il progettista (o meglio il team di progettisti) è quindi costretto a intraprendere una serie di costose iterazioni di sintesi, seguite dalla generazione del layout, fino a ottenere una soluzione accettabile, cioè che soddisfi tutti i vincoli temporali (Figura 8-17b e Figura 8-17c). Ognuna di queste iterazioni può richiedere diversi giorni, mentre la sintesi della rete di connessioni di un chip complesso può richiedere una settimana su uno dei computer più avanzati. Il numero di iterazioni richieste continua a crescere al ridursi delle geometrie minime della tecnologia. Questo problema, chiamato *timing closure*, ha reso evidente la necessità di nuove soluzioni e di cambiamenti nelle metodologie di progetto.

La risposta comune consiste nel creare un'integrazione più stretta tra i processi di sintesi logica e fisica. Per esempio, se lo strumento di sintesi logica realizza anche una parte del piazzamento (o quantomeno indirizza la fase di piazzamento) si possono ottenere stime più precise dei parametri di layout. La Figura 8-18 mostra un esempio di un ambiente di progettazione che fonde la sintesi RTL con un piazzamento e collegamento del primo ordine. La netlist che ne risulta è poi passata a uno strumento di ottimizzazione che realizza il piazza-

Materiale protetto da copyright



(a) Cella gate-array primitiva

(b) Cella programmata, che realizza una NOR a quattro ingressi

Figura 8-19 Un esempio dell'approccio a gate-array.

8.5.1 Matrici logiche senza interconnessioni

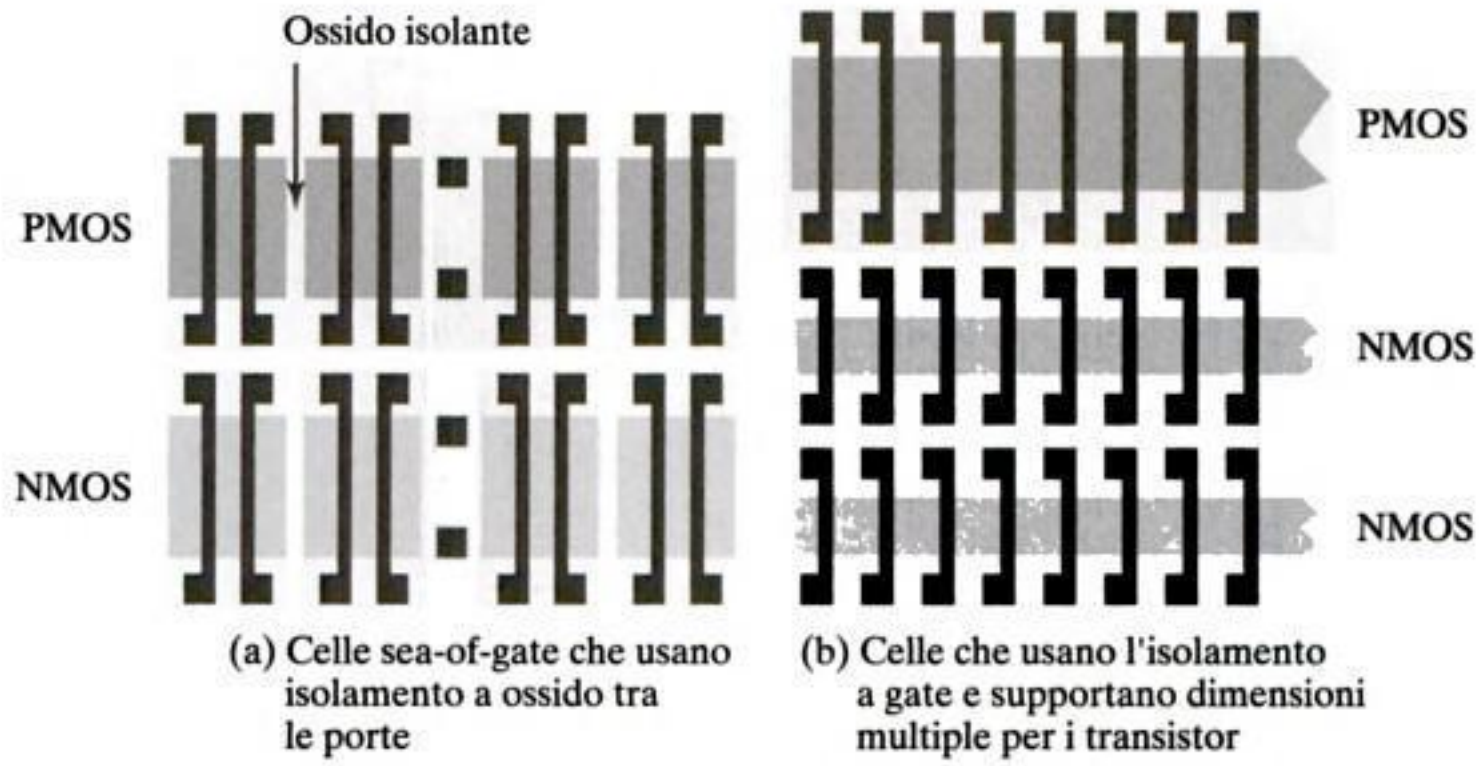
In questo approccio, lotti di wafer che contengono matrici di celle primitive o di transistor vengono fabbricati e immagazzinati. Tutti i passi di fabbricazione necessari per realizzare i transistor sono standardizzati e indipendenti dall'applicazione finale. Per trasformare questi wafer non commissionati in un sistema effettivo, è necessario aggiungere solo le interconnessioni che si desiderano, determinando la funzionalità globale del chip con la semplice definizione di pochi passi per la deposizione dei metalli. Questi livelli possono essere progettati e applicati ai wafer precostruiti molto più rapidamente, riducendo il tempo di ritorno del circuito a una settimana o meno.

Questo approccio è spesso indicato con il termine *gate-array* o *sea-of-gates*, a seconda della struttura del wafer prefabbricato. Per illustrare il concetto, consideriamo la cella primitiva a gate-array mostrata nella Figura 8-19a, che comprende quattro transistor NMOS, quattro transistor PMOS, le connessioni ai gate in polisilicio e le linee di massa e alimentazione. Vi sono due punti di contatto disponibili per ogni area di diffusione e due possibili punti di connessione per ogni linea di polisilicio. Possiamo trasformare questa cella, che per ora non realizza nessuna funzione logica, in un circuito reale aggiungendo alcune piste sul livello metallico e sui fori di contatto. Questo è mostrato nella Figura 8-19b, dove la cella è trasformata in una porta NOR a quattro ingressi.

L'approccio a *gate-array* originale[2] piazza le celle in righe, separate da canali per le connessioni, come mostra la Figura 8-20a; l'apparenza globale è simile a quella della tecnica tradizionale a celle standard. Con l'avvento dei livelli di metallizzazione aggiuntivi, i canali di connessione possono essere eliminati e i collegamenti possono essere realizzati al di sopra delle celle primitive, lasciando ogni tanto delle celle inutilizzate. Questa architettura senza canali, detta anche *sea of gates* (Figura 8-20b), comporta una maggiore densità e permette di raggiungere l'integrazione di milioni di porte su uno stesso chip. Un altro vantaggio

[2]Questo approccio è spesso detto *channeled gate array*.

Materiale protetto da copyright



(a) Celle sea-of-gate che usano isolamento a ossido tra le porte

(b) Celle che usano l'isolamento a gate e supportano dimensioni multiple per i transistor

Figura 8-21 Esempi di celle primitive di sea-of-gate (da [Veendrick92]).

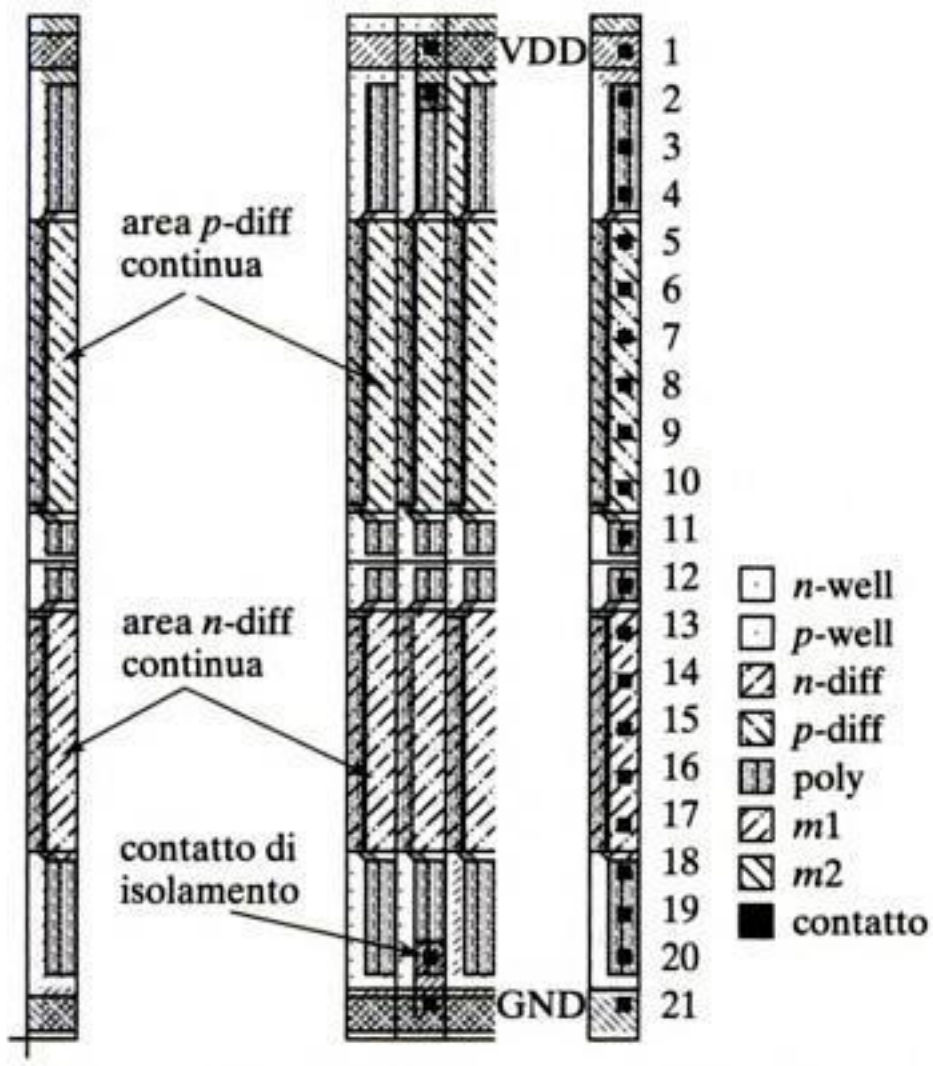


Figura 8-22 Cella fondamentale di gate-array con isolamento a gate (da [Smith97]).

Vale la pena di osservare che la cella della Figura 8-21b mette a disposizione due righe di transistor NMOS più piccoli, che possono essere connessi in parallelo se necessario. Transistor di dimensione ridotta sono utili per costruire porte logiche a pass-transistor o celle di memoria. Poiché il gate-array è programmato configurando le interconnessioni, i ritardi di propagazione sono dominati dalle capacità delle piste. Questo fatto sembra far preferire dispositivi con dimensioni maggiori, che però occupano un'area maggiore. D'altra parte è possibile anche realizzare transistor più grandi connettendo in parallelo alcuni dispositivi più piccoli.

Materiale protetto da copyright

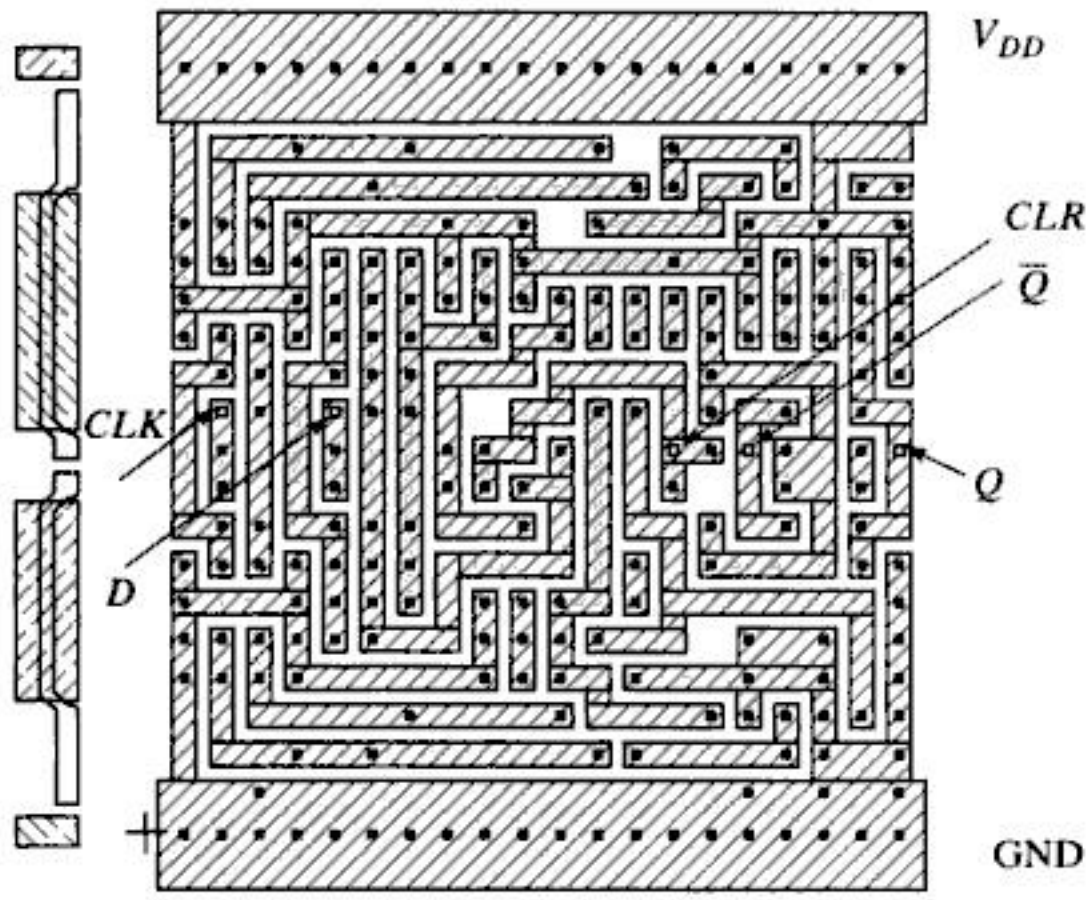


Figura 8-23 Flip-flop implementato con gate-array con isolamento a gate. La cella base è mostrata a sinistra (da [Smith97]).

La mappatura di un circuito logico in un array di celle è un processo largamente automatizzato, che coinvolge la sintesi logica, seguita da piazzamento e connessione. La qualità di questi strumenti ha un impatto notevole sulla densità e sulle prestazioni finali del sistema. La percentuale di sfruttamento dei dispositivi nelle strutture a sea-of-gates dipende fortemente dal tipo di applicazione per cui sono usate. In strutture regolari quali le memorie si può ottenere uno sfruttamento prossimo al 100%. Per altre applicazioni, il fattore di uso può essere sostanzialmente minore (< 75%), principalmente a causa dei vincoli posti dalle connessioni. La Figura 8-23 mostra l'esempio di una macrocella di un flip-flop, realizzata in una libreria di gate array a isolamento di gate.

Similmente alla situazione spiegata nel caso della progettazione a celle standard, i progettisti di sea-of-gates hanno presto scoperto che sistemi con molte porte logiche tipicamente richiedono anche grandi quantità di memoria. Realizzare fisicamente queste celle di memoria al di sopra delle celle fondamentali dei gate-array è possibile, ma non molto efficiente. Un approccio più efficiente è quello di riservare una parte di area a moduli di memoria dedicati. La combinazione di gate array con macro celle fisse è detto *embedded gate-array*. Altri moduli quali microprocessori e microcontrollori sono pure degli ottimi candidati a essere inclusi.

Esempio 8.7 Realizzazione di un chip a sea-of-gates

Un esempio di implementazione a sea-of-gates è mostrato nella Figura 8-24. La matrice ha una capacità massima di 300 000 porte ed è realizzata in una tecnologia CMOS da 0.6 μm. La parte in basso a sinistra della matrice realizza una memoria, da cui risulta un layout regolare e modulare. Il resto della matrice realizza della logica generica.

Materiale protetto da copyright

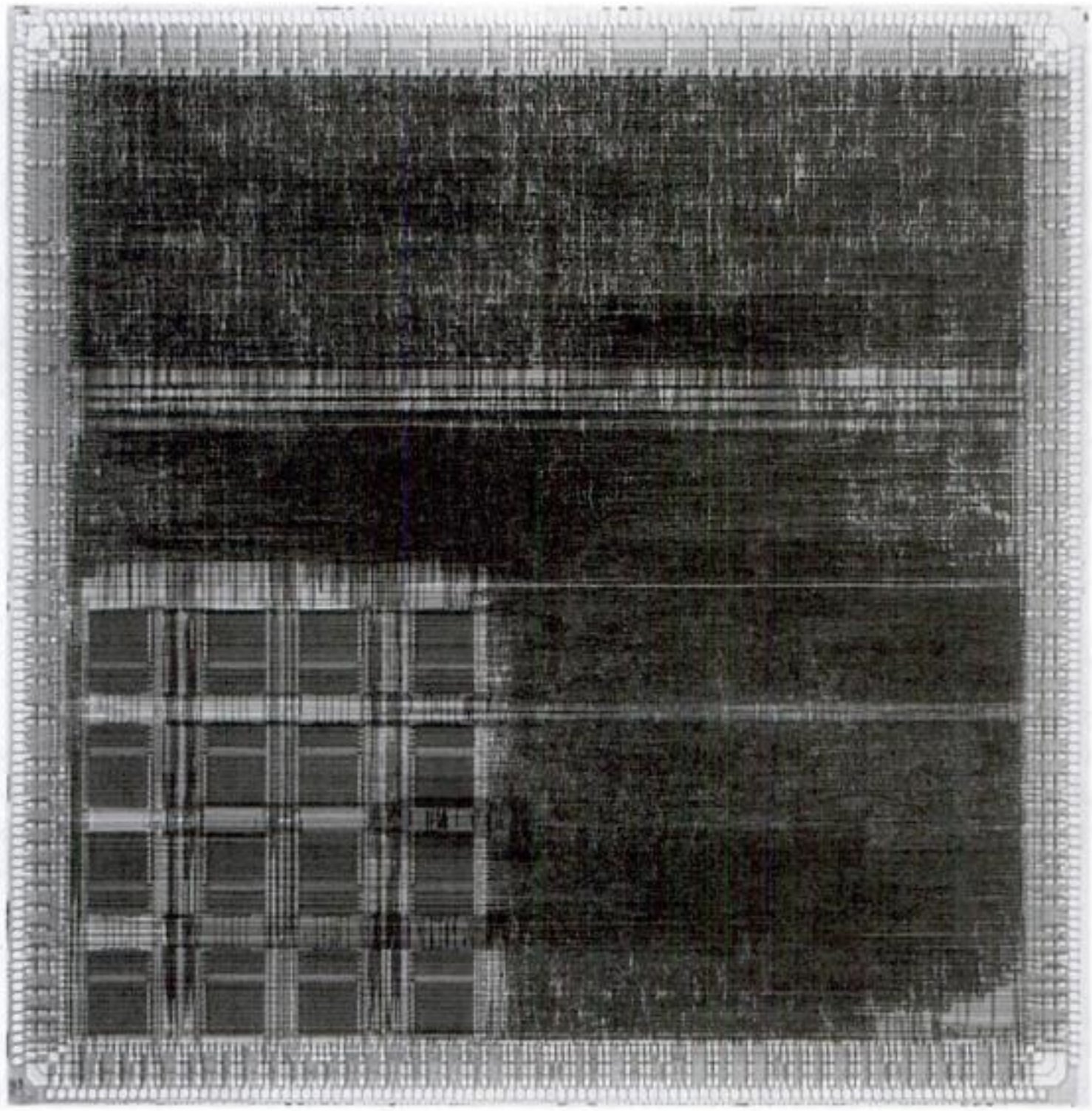


Figura 8-24 Fotografia di un chip a gate-array (LEA300K) (*Gentile concessione di LSI Logic*).

Scelte progettuali—Gate array o celle standard

Negli anni '80 e '90, quando la maggioranza dei chip conteneva meno di 50 000 porte, i cicli di progetto potevano durare settimane o al più qualche mese. Il risparmio di tempo di due o tre settimana ottenuto usando i gate-array rappresentava quindi un guadagno significativo, più che sufficiente a giustificare l'aumento dell'area del chip. Con gli odierni processi submicrometrici e le complessità dell'ordine di molti milioni i tempi di progetto si sono allungati e il tempo necessario alla fabbricazione non è più un problema cruciale. Inoltre la metallizzazione è divenuta il passo che richiede più tempo e anche quello che può influire maggiormente sulla resa del processo di fabbricazione, riducendo ulteriormente i vantaggi che i gate array hanno da offrire. Di conseguenza i gate array hanno perso gran parte della loro popolarità. Un'altra alternativa per una prototipazione veloce, le matrici con interconnessioni programmabili di cui si parla nel prossimo paragrafo, è stata introdotta e ha sostituito i gate array in una porzione significativa del mercato.

Ciò nonostante non bisogna pensare ai moduli logici a maschere programmabili come una cosa che appartiene al passato. Uno stile di layout fisso e regolare ha il vantaggio di rendere possibile una stima facile e accurata dei fattori di carico, dei parassiti delle interconnessioni e del rumore di accoppia-

Materiale protetto da copyright

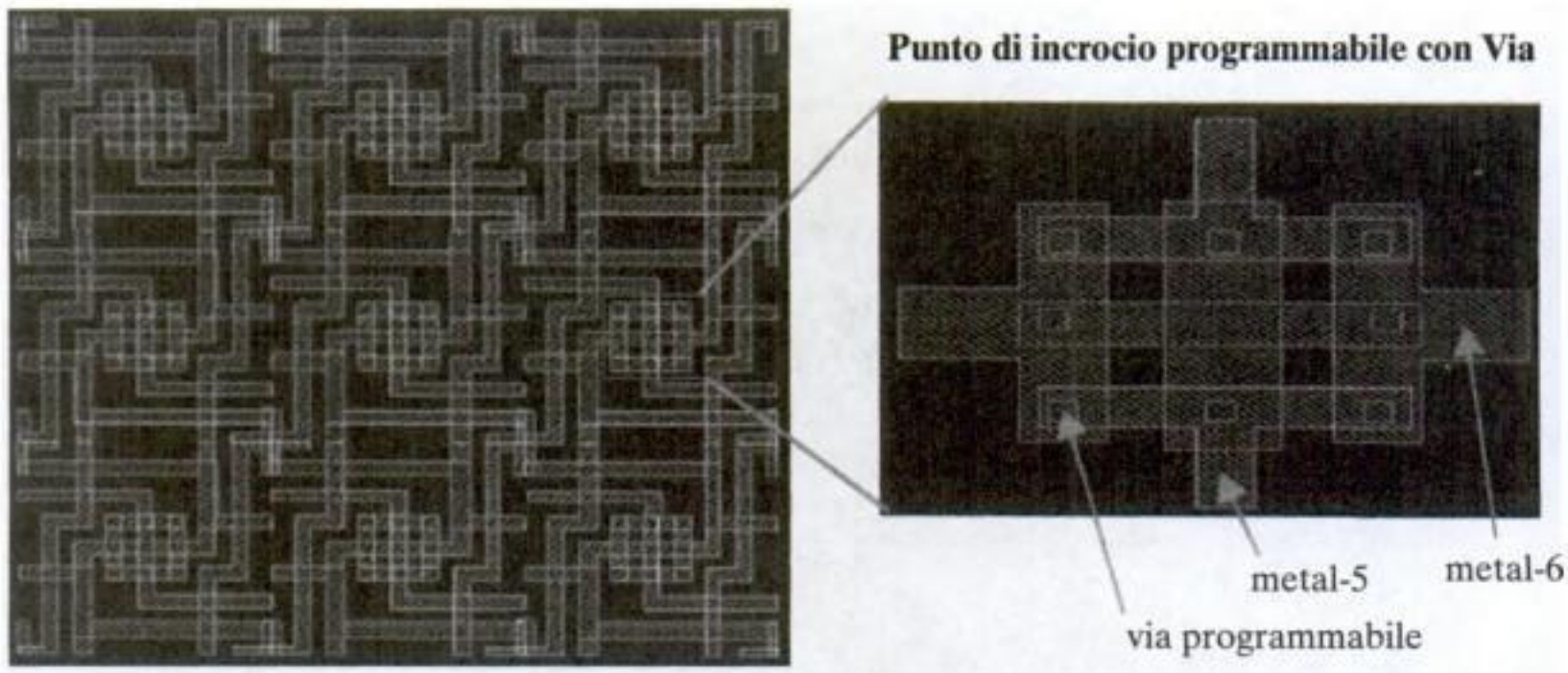


Figura 8-25 Gate array programmabile con via. I via sono usati per vincolare una rete di interconnessioni generica a una funzione specifica, ottenendo delle matrici prevedibili[Pileggi02].

mento. Ciò è in contrasto con l'approccio a celle standard dove questi valori sono in realtà noti solo dopo il piazzamento, la stesura delle connessioni e l'estrazione. Si può pensare di riempire parti di chip grandi con matrici regolari di porte logiche, composte da celle logiche neutre (prediffuse) ricoperte da una griglia di connessioni. La programmazione del modulo si realizza inserendo delle *via* in alcune posizioni predefinite. Come mostra la Figura 8-25, l'uso di *via* programmabili nei punti di incrocio permette di mappare una gran varietà di configurazioni di connessioni su una griglia di connessioni regolare e ripetitiva. Riteniamo quindi che le matrici prediffuse sopravvivranno ancora. ■

8.5.2 Matrici a interconnessioni programmabili

Sebbene le matrici prediffuse offrano una strada rapida all'implementazione, sarebbe ancora più efficiente poter evitare del tutto i passi del processo dedicati al particolare sistema progettato. Questo porta al concetto di chip preprocessati che possano essere programmati sul campo (cioè al di fuori della fonderia) per realizzare un dato insieme di funzioni booleane. Tale matrice di celle programmabili e pre-connesse è detto *FPGA* (*field-programmable gate array*) e ha il vantaggio di separare la fase di fabbricazione dalla quella di implementazione e può quindi essere ammortizzato su un numero molto vasto di progetti. L'implementazione può essere svolta direttamente dall'utente, con tempi di fabbricazione praticamente nulli. Lo svantaggio principale di questa tecnica è la perdita in termini di prestazioni e densità di integrazione rispetto agli approcci più dedicati.

È necessario affrontare due questioni principali per implementare un insieme di funzioni booleane su una matrice regolare di celle senza affidarsi ad alcun passo di processo:

1. Come possiamo costruire logica "programmabile", cioè una logica che deve poter realizzare qualsiasi funzione booleana?
2. Come e dove immagazziniamo il *programma* (detto anche *configurazione*) che vincola la matrice programmabile a una funzione logica specifica?

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

ponente. Per programmare il componente all'avvio, i dati di programmazione sono trasferiti in modo seriale nel componente attraverso una sola linea. Ai fini pratici, durante questa fase si può interpretare la RAM della FPGA in fase di configurazione come un enorme registro a scorrimento. Una volta caricate tutte le celle di memoria, comincia il funzionamento regolare. Il tempo di configurazione è proporzionale al numero di elementi programmabili e può essere proibitivo nelle odierne FPGA, che spesso realizzano più di un milione di porte. I componenti più recenti si affidano quindi sempre più a interfacce di programmazione parallele che permettono di programmare più celle contemporaneamente.

Al contrario della controparte non volatile, le FPGA volatili non richiedono passi di processo speciali e possono essere realizzate con processi CMOS standard. Inoltre i progettisti possono riusare i chip durante la prototipazione; la logica può essere modificata o aggiornata anche sul campo e un cliente può ricevere un nuovo file di configurazione per aggiornare il chip, piuttosto che dover ricevere un nuovo chip. Inoltre la logica può anche essere modificata dinamicamente, durante l'esecuzione. Quest'ultimo approccio è detto *riconfigurazione* ed era piuttosto famoso alle fine degli anni '90. Da un certo punto di vista, possiamo dire che viene sfruttato a livello di progettazione logica, un approccio che ha decretato il grande successo del dominio della programmazione (rappresentato per eccellenza dai microprocessori).

Per quanto riguarda la prima domanda, la risposta è in un certo senso più vasta. Implementare un circuito complesso in modo programmabile richiede che sia le funzioni logiche, sia le interconnessioni tra queste siano realizzate in modo programmabile. Nei prossimi paragrafi, esamineremo dapprima diversi modi per realizzare della logica programmabile e quindi daremo una panoramica delle interconnessioni programmabili. Infine vedremo nel dettaglio un certo numero di modi specifici per mettere insieme i due aspetti.

Logica programmabile

Come nel caso della progettazione semi-automatica, sono attualmente in uso due approcci fondamentalmente diversi alla logica programmabile: a *matrice* e a *celle*.

Logica programmabile a matrice Abbiamo già discusso in precedenza di come una PLA implementi funzioni logiche booleane in modo regolare (vedi pagina 415). Un approccio simile può essere usato anche per le FPGA. Consideriamo, per esempio, la struttura logica della Figura 8-27. Un cerchio (o) a una intersezione sta a indicare una connessione programmabile, cioè un punto di interconnessione che può essere abilitato o meno. Osservando il diagramma ci si rende conto che questo è equivalente a una PLA, in cui sia il piano AND che quello OR possono essere programmati selettivamente abilitando le connessioni. Questo approccio permette di realizzare funzioni logiche generiche nella forma a due livelli di *somma di prodotti*. Il piano AND realizza i mintermini richiesti, mentre il piano OR realizza la somma di un dato insieme di prodotti per generare l'uscita. Per includere in un mintermine una variabile di ingresso specifica (per esempio I_1), basta chiudere l'interruttore all'intersezione tra il segnale di ingresso e il mintermine. In modo analogo, un mintermine è incluso in

Materiale protetto da copyright

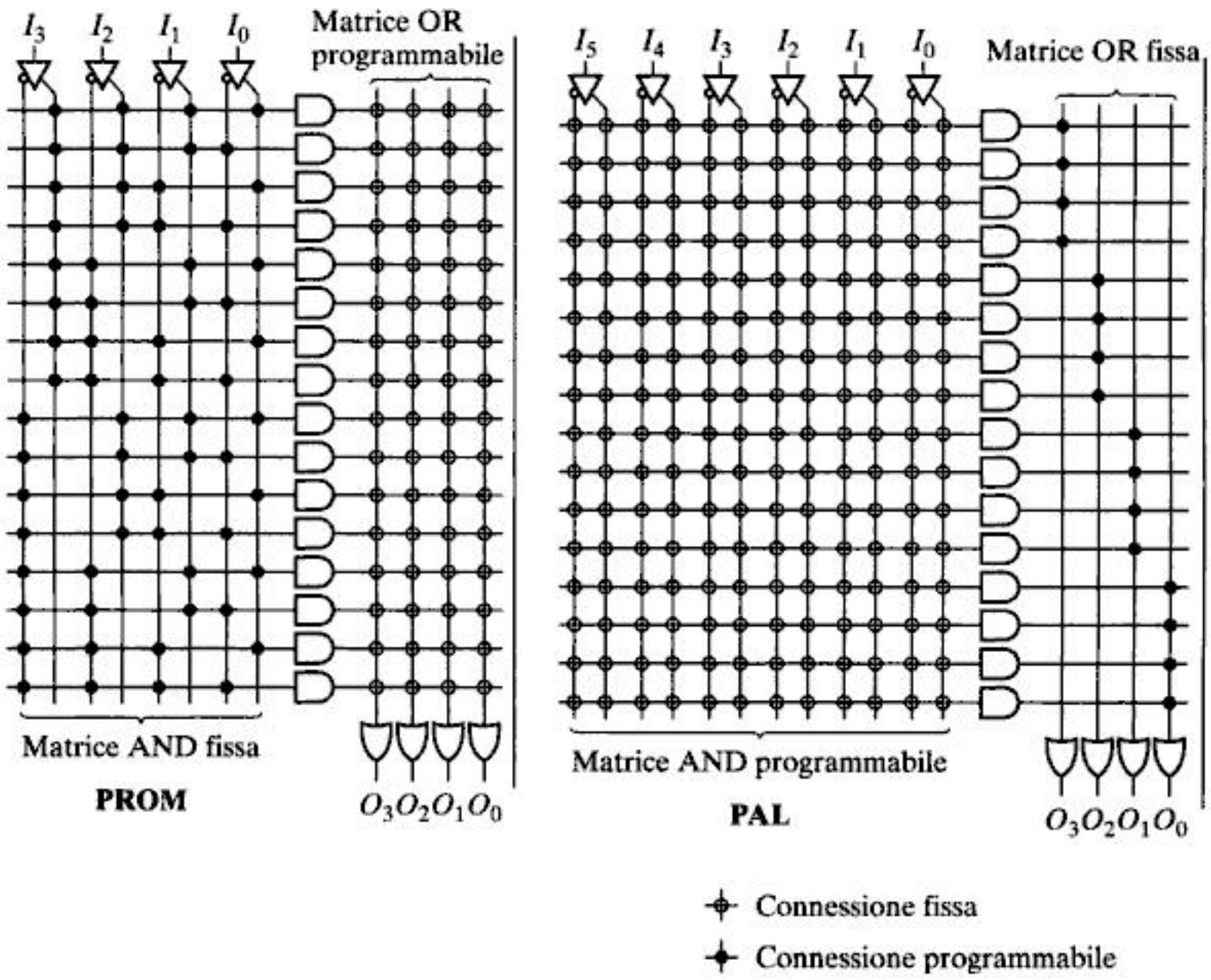


Figura 8-28 Dispositivi logici programmabili (PLD) alternativi a fusibili.

1. Partizionare la matrice in un certo numero di sezioni più piccole, spesso dette macrocelle.
2. Introdurre flip-flop e fornire la possibilità di realizzare un percorso di retroazione dai segnali di uscita a quelli di ingresso.

Un esempio di come ciò si possa ottenere è mostrato nella Figura 8-29. La PAL è formata da k macrocelle, ciascuna delle quali può selezionare tra i ingressi e generare al massimo j termini prodotto. Ciascuna macrocella contiene un unico registro programmabile, che può essere configurato come un flip-flop *D*, *T*, *J-K*, o *S-R* sincrono. I k segnali di uscita sono riportati al bus di ingresso e quindi sono un sottoinsieme degli i segnali di ingresso.

L'approccio a PLA per la logica configurabile ha due vantaggi peculiari:

- La struttura è molto regolare, rendendo la stima dei parassiti piuttosto semplice e permette una previsione molto precisa di area, velocità e dissipazione di potenza.
- Fornisce un'implementazione efficiente per funzioni logiche adatte a essere rappresentate a due livelli, come per esempio le funzioni con fan-in elevato e le macchine a stati finiti usate nei controllori.

D'altra parte la struttura a matrice ha lo svantaggio di un'elevata ridondanza. Ciascun nodo intermedio ha una capacità apprezzabile che influisce negativamente sulla velocità e sul con-

Materiale protetto da copyright

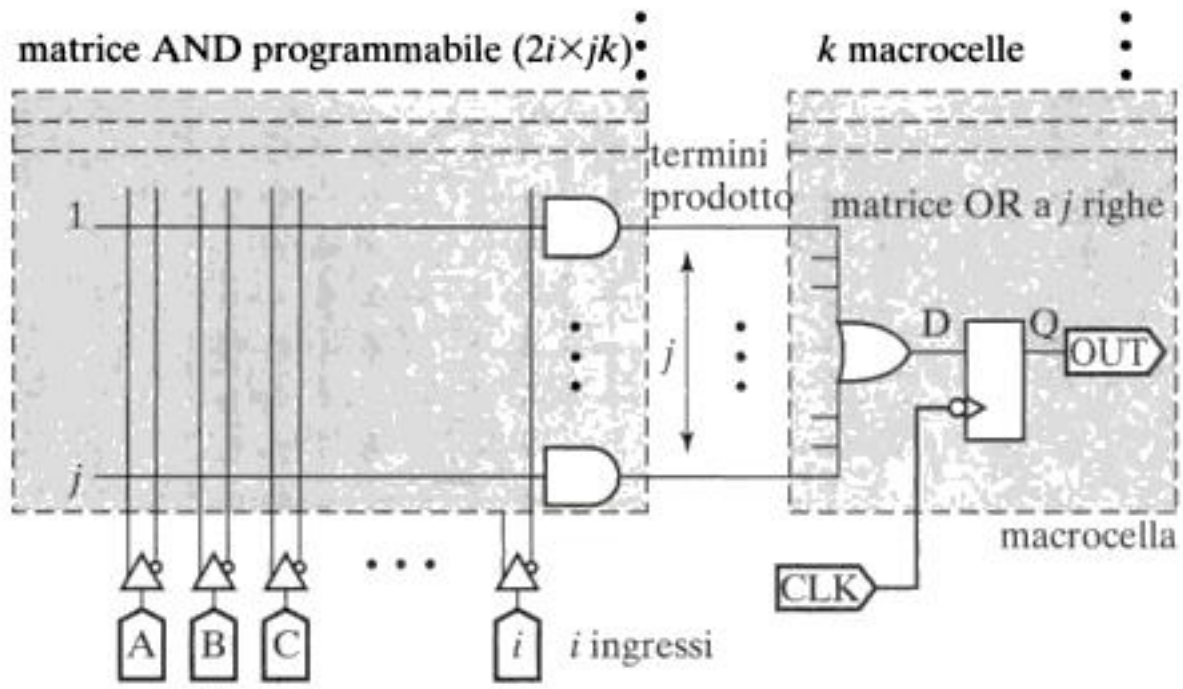


Figura 8-29 Schema circuitale di una PAL con *i* ingressi, *j* mintermini e *k* macrocelle (o uscite) [Smith97].

sumo di potenza. Questo è particolarmente vero quando parti della matrice sono poco sfruttate, quando per esempio solo alcuni dei mintermini sono effettivamenti necessari per la valutazione della funzione logica.

Esempio 8.8 Un esempio di macrocella programmabile

La Figura 8-30 mostra un esempio di come programmare un modulo PROM. La struttura è programmata per realizzare le funzioni logiche usate in precedenza durante l'esame delle PLA (Eq. (8.1)):

$$f_0 = x_0 x_1 + \overline{x_2}$$

$$f_1 = x_0 x_1 x_2 + \overline{x_2} + \overline{x_0} x_1$$

Osserviamo che solo una frazione delle matrice è usata, poiché il numero variabili di ingresso (3) e di uscita (2) è minore delle dimensioni della matrice, cioè 4×4. Le variabili di ingresso inutilizzate sono vincolate a 0 o a 1. I grossi punti disegnati sul piano OR rappresentano i nodi programmati. Il lettore è invitato a ripetere l'esercizio per i moduli PLA e PAL presentati nella Figura 8-28 e nella Figura 8-29.

Logica programmabile a celle L'implementazione di una funzione logica come somma di prodotti si traduce in strutture regolari ed è molto efficace per funzioni logiche a molti ingressi, come le macchine a stati finiti, ma ha prestazioni piuttosto scarse per funzioni logiche con grande fan-out o che sono realizzate più convenientemente a più livelli (per esempio, gli operatori somma e prodotto). Per queste funzioni, si possono impiegare altre strutture più in linea con la natura a livelli multipli, come già fatto nel caso delle celle standard e dei sea-of-gates.

Vi sono molti modi per progettare un blocco logico che possa essere configurato per realizzare una vasta gamma di funzioni logiche. Una possibilità è di usare dei *multiplexer*

Materiale protetto da copyright

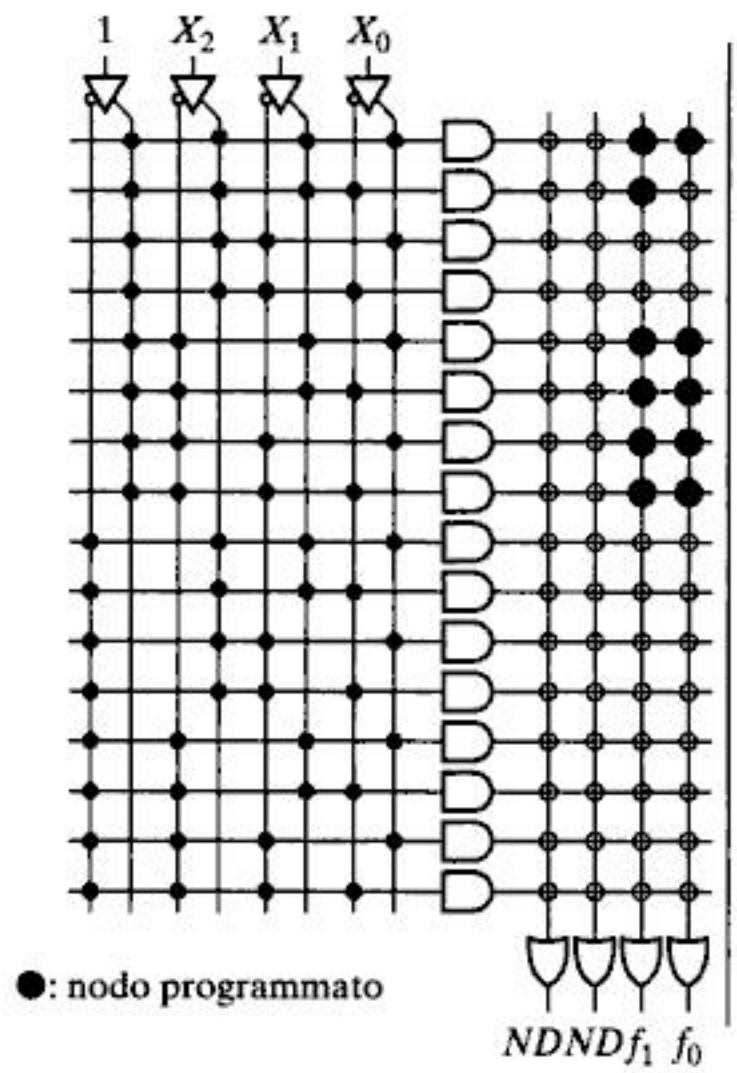


Figura 8-30 Programmare una PROM.

come generatori di funzione. Consideriamo il multiplexer a due ingressi della Figura 8-31, che realizza la funzione logica F:

$$F = A \cdot \bar{S} + B \cdot S \tag{8.3}$$

Scegliendo con cura la connessione tra le variabili X e Y e i terminali di ingresso A, B e S, possiamo programmare il multiplexer per realizzare dieci funzioni logiche distinte a uno o più ingressi (vedi Figura 8-31).

Possiamo combinare alcuni multiplexer per formare porte logiche più complesse. Consideriamo, per esempio, la cella logica della Figura 8-32, che è usata nelle FPGA della famiglia Actel ACT e consiste di tre multiplexer e di una porta NOR con due ingressi ciascuno. La cella può essere programmata per realizzare qualsiasi funzione booleana a due ingressi, alcune funzioni a tre ingressi o un latch.

Esempio 8.9 Una cella programmabile

Possiamo verificare che la cella logica della Figura 8-32 agisce come una porta XOR a due ingressi (*In1* e *In2*), con le condizioni di programmazione seguenti. Ipotizziamo che i multiplexer selezionino il segnale di ingresso inferiore quando il segnale di controllo è alto. Poniamo quindi:

$$A = 1;\ B = 0;\ C = 0;\ D = 1;\ SA = SB = In1;\ S0 = S1 = In2$$

Materiale protetto da copyright

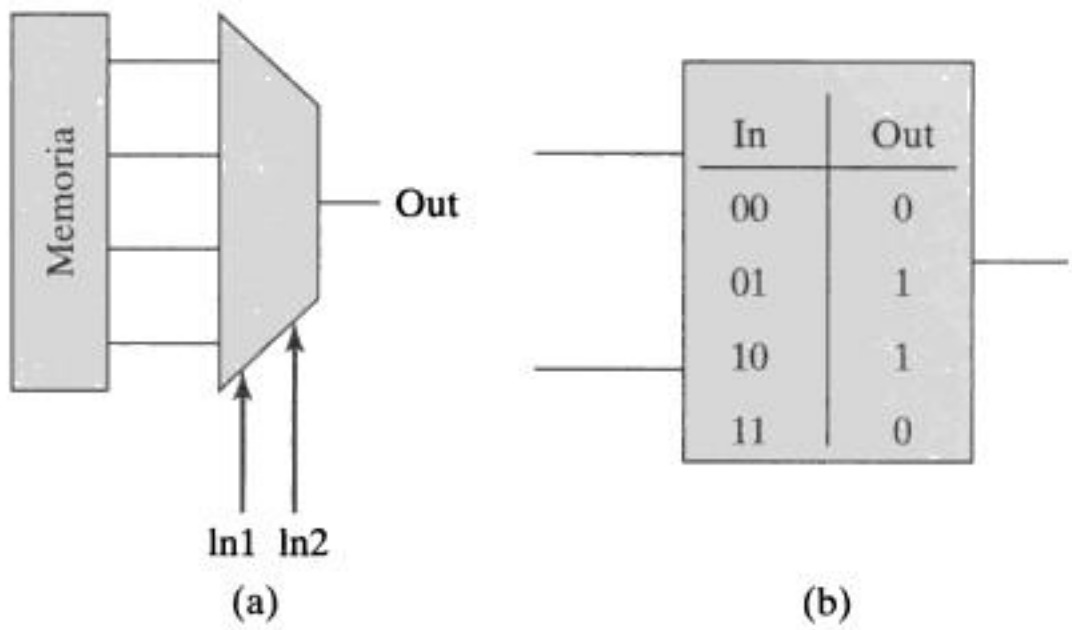


Figura 8-33 Cella logica configurabile a lookup table. (a) Schema circuitale della cella; (b) programmazione della cella per realizzare una funzione XOR.

detta la lookup table, in modo che contenga la tabella di verità di tale funzione. Le variabili di ingresso fungono da segnali di controllo di un multiplexer, che preleva il valore appropriato dalla memoria. L'idea è illustrata nella Figura 8-33 per una cella a due ingressi. Per implementare la funzione XOR, la lookup table memorizza la colonna di uscita della tabella di verità della funzione XOR, cioè "0 1 1 0". Per un valore in ingresso pari a "0 0", il multiplexer seleziona il primo valore nella tabella ("0"), e così via. Con questo approccio, qualsiasi funzione a due ingressi può essere realizzata tramite una semplice (ri)programmazione della memoria.

Come nel caso dell'approccio a multiplexer, si possono anche costruire porte più complesse. Questo si ottiene o combinando un certo numero di LUT, o aumentando la dimensione della LUT o tramite la combinazione di questi due sistemi. Incorporando dei flip-flop, si possono ottenere funzionalità ulteriori.

Esempio 8.10 Cella logica programmabile a LUT

La Figura 8-34 mostra il *blocco logico configurabile* (CLB, *Configurable Logic Block*), che costituisce la cella elementare di una FPGA della serie 4000 della Xilinx [Xilinx4000]. Essa combina due LUT a quattro ingressi che pilotano una LUT a tre ingressi. La cella dispone anche di due flip-flop, i cui ingressi possono essere una qualsiasi delle uscite delle LUT F, G o H, o un ingresso esterno addizionale D_{in}, e le cui uscite sono disponibili ai pin di uscita XQ e YQ. Le uscite X e Y forniscono le uscite delle LUT e rendono possibile la costruzione di funzioni combinatorie più complesse. La cella ha quattro ingressi addizionali ($C1 \ldots C4$) che possono essere usati indifferentemente come ingressi o come segnali di set/reset e di sincronia per i flip-flop.

Interconnessioni programmabili

Finora, abbiamo discusso in dettaglio delle soluzioni per realizzare circuiti logici programmabili. Una questione pressante è ora come trasformare in elementi configurabili anche le interconnessioni tra le celle. Per sfruttare fino in fondo le celle logiche disponibili, la rete di

Materiale protetto da copyright

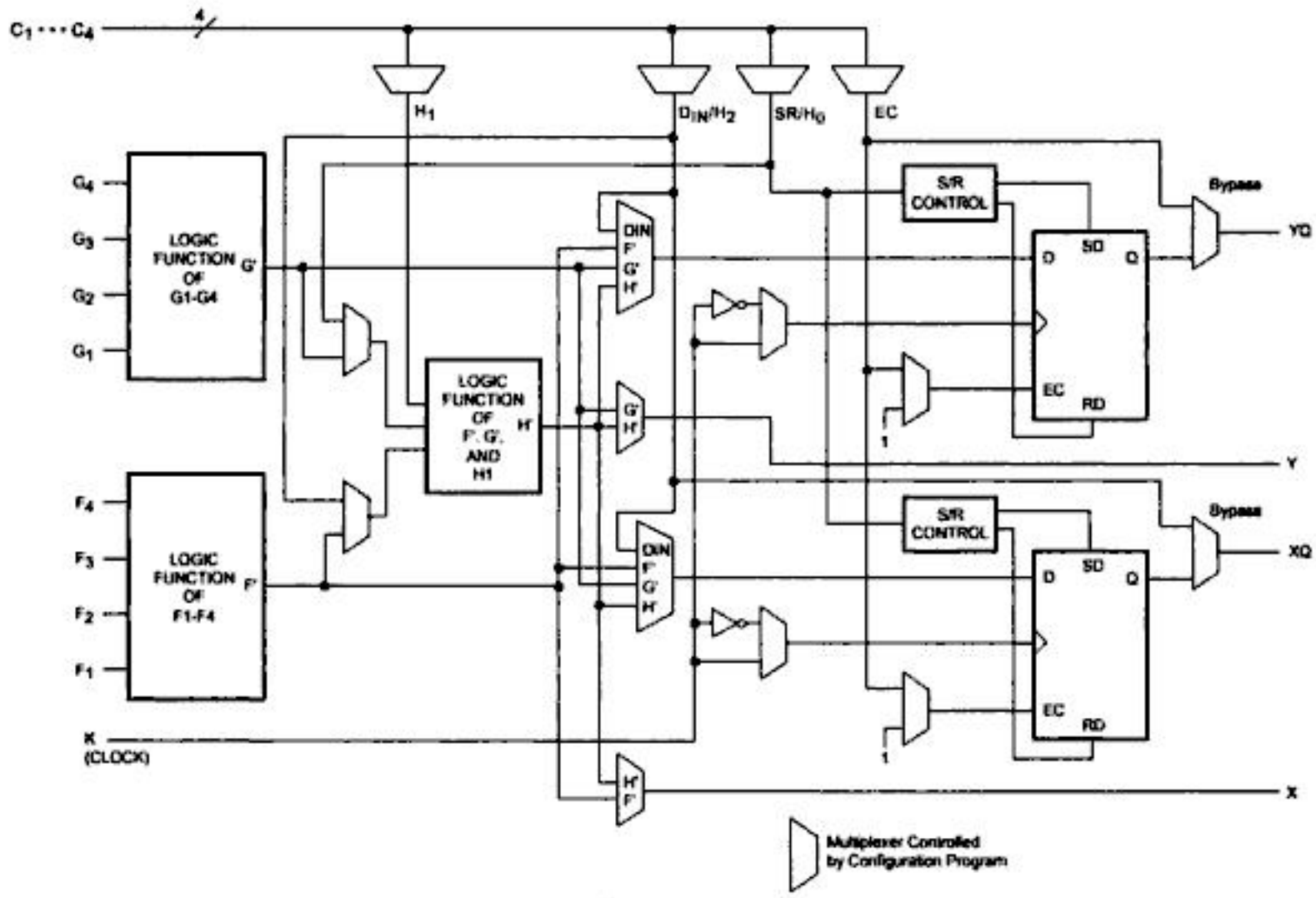


Figura 8-34 Schema a blocchi semplificato del blocco CLB della serie XC4000 (la RAM e le funzioni di carry-logic non sono mostrate) [Xilinx4000].

interconnessioni deve essere flessibile e si devono evitare colli di bottiglia dovuti alla definizione delle connessioni. La velocità è un altro requisito importante, perché i ritardi delle connessioni tendono a essere il fattore dominante in questi stili di progettazione. Allo stesso tempo, il lettore dovrebbe essere consapevole del fatto che le interconnessioni programmabili si pagano pesantemente in termini area, prestazioni e potenza. Di fatto la maggior parte della dissipazione di potenza nelle FPGA è attribuibile alla rete di interconnessioni [George01].

Anche rispetto alla configurabilità delle interconnessioni, possiamo distinguere tra tecnologie realizzative basate sulla definizione di un numero limitato di maschere del processo di fabbricazione e tecnologie che prevedono la configurazione dopo la fabbricazione del cricuito. Tra queste ultime distinguiamo anche tra interconnessioni programmabili una sola volta e quelle riprogrammabili. Un altro elemento di distinzione sussiste tra connessioni locali e cella-a-cella e segnali globali quali i clock, che devono essere distribuiti attraverso tutto il chip, con ritardi minimi. Nella classe delle connessioni programmabili locali, possiamo distinguere due gruppi principali: connessioni programmabili a matrice e a commutazione.

Connessioni programmabili a matrice In questo approccio, le connessioni sono raggruppate in appositi canali, ciascuno dei quali contiene un insieme completo di piste orizzontali e verticali. Una pista di interconnessione può quindi essere programmata cortocircuitando alcune delle intersezioni tra piste verticali e orizzontali (vedi Figura 8-35). Questo si può ottenere disponendo un pass transistor a ciascun punto di intersezione: chiudere la connessione significa portare a livello alto il segnale di controllo, programmando un "1" nella cella

Materiale protetto da copyright

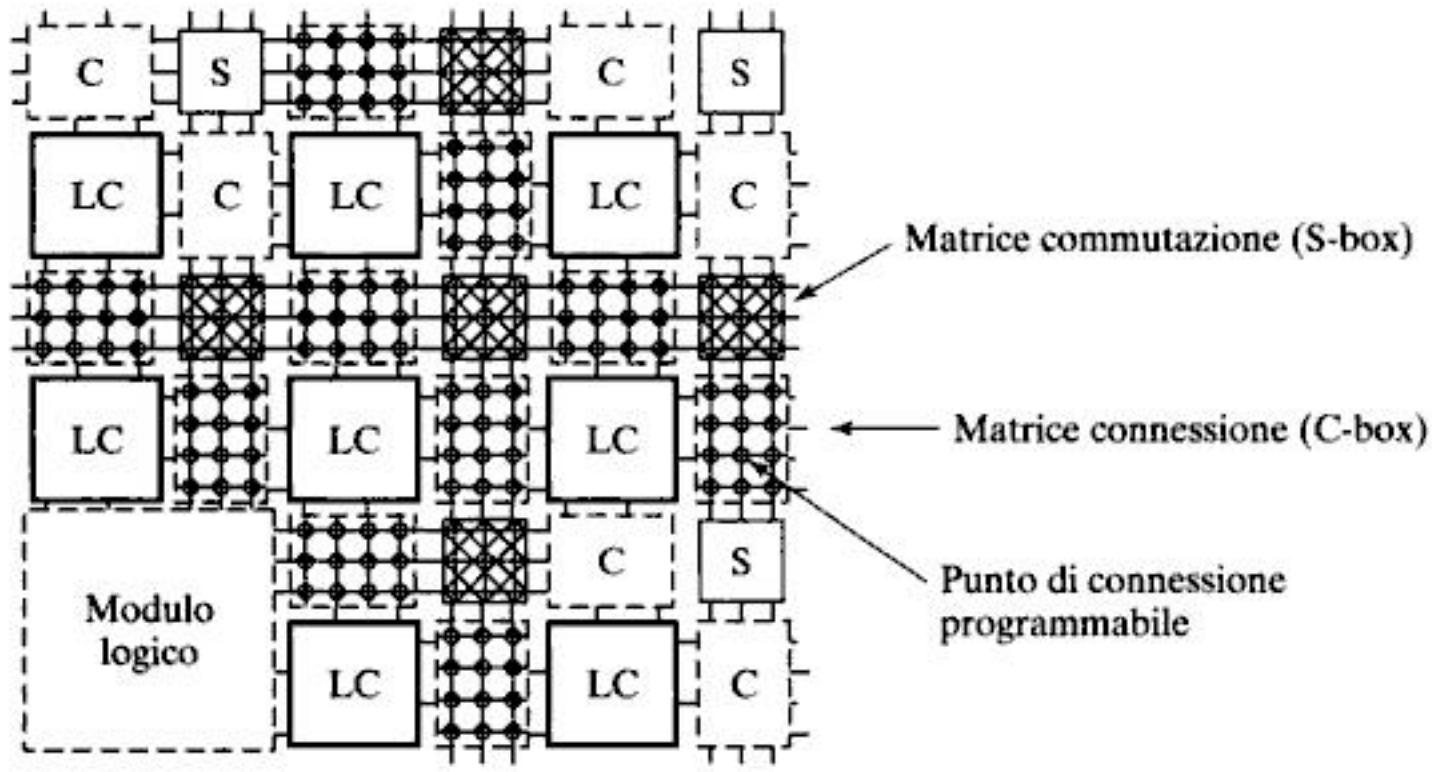


Figura 8-37 Interconnessioni programmabili a rete (*gentile concessione di Andre Dehon e John Wawrzyniek*).

sono possibili correzioni o estensioni del circuito ed è necessario usare nuovi componenti a ogni modifica del sistema. Una vera (ri)programmabilità richiede una strategia di connessione più efficiente.

Connessioni programmabili a commutazione Avendo compreso che garantire la possibilità di realizzare qualsiasi configurazione di interconnessione tra tutte le piste del circuito non è richiesto da nessun sistema realistico, è possibile individuare un approccio più efficiente. Riducendo il numero di piste di connessione e di punti di contatto, è comunque possibile realizzare tutte le connessioni necessarie, diminuendo decisamente la ridondanza. Lo svantaggio di tale approccio è che qualche volta non è possibile realizzare una connessione. Molto spesso però, questa situazione si risolve con modifiche a livello di sistema, per esempio scegliendo un altro gruppo di celle logiche per una data funzione.

Buona parte delle connessioni locali possono essere realizzate prevedendo una struttura di interconnessione a rete per le celle adiacenti. Come si vede nella Figura 8-37, le uscite di ciascuna cella logica (LC) sono distribuite alle celle vicine verso nord, est, sud e ovest. Per realizzare le connessioni tra celle non contigue e per distribuire lungo tutto il chip i segnali globali, è necessario inserire degli appositi canali per ospitare un certo numero di piste di interconnessione generiche. In corrispondenza dell'incrocio tra piste orizzontali e verticali, sono disponibili della matrici di commutazione (*S-box*) programmabili mediante una memoria RAM per instradare i dati. Gli ingressi e le uscite delle celle sono connesse alle interconnessioni globali con contatti programmabili (*C-box*) mediante RAM. La Figura 8-38 mostra un ingrandimento, evidenziando i pass-transistor usati nei blocchi di commutazione per realizzare i contatti programmabili. Teniamo presente che la realizzazione a pass-transistor singoli implica la degradazione del valore logico trasmesso. Sebbene questo sia conveniente dal punto di vista del consumo, l'escursione ridotta del segnale ha conseguenze negative in termini di prestazioni. Potrebbe quindi essere richiesto l'impiego di dispositivi a soglia nulla, level restorer o segnali di controllo a tensione più elevata.

Materiale protetto da copyright

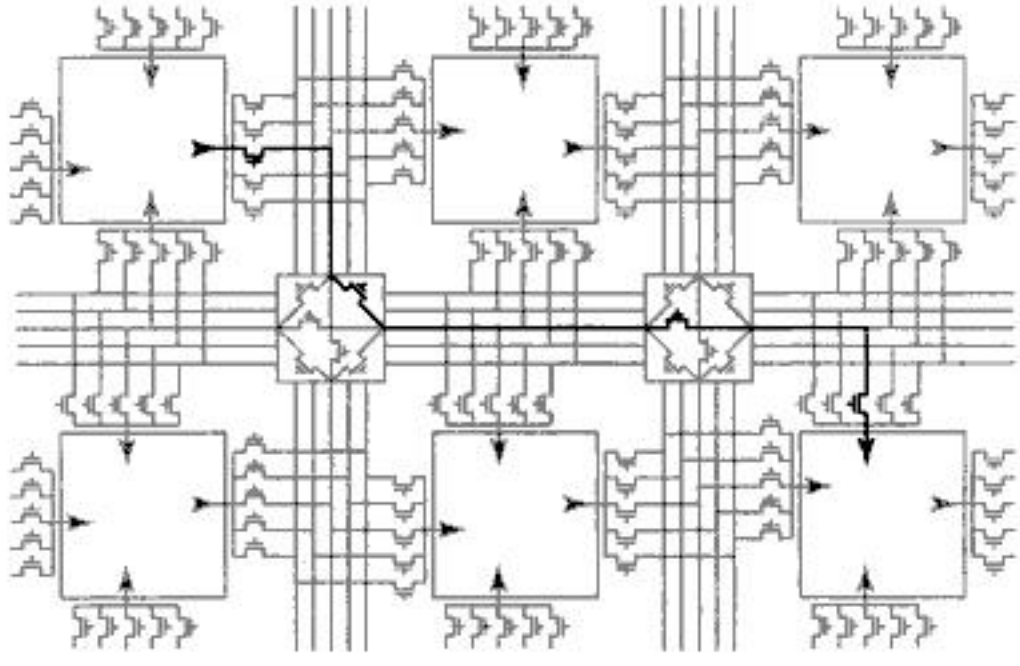


Figura 8-38 Schema circuitale a livello di transistor di un sistema di interconnessioni con struttura a rete (*gentile concessione di Andre Dehon e John Wawrzyniek*).

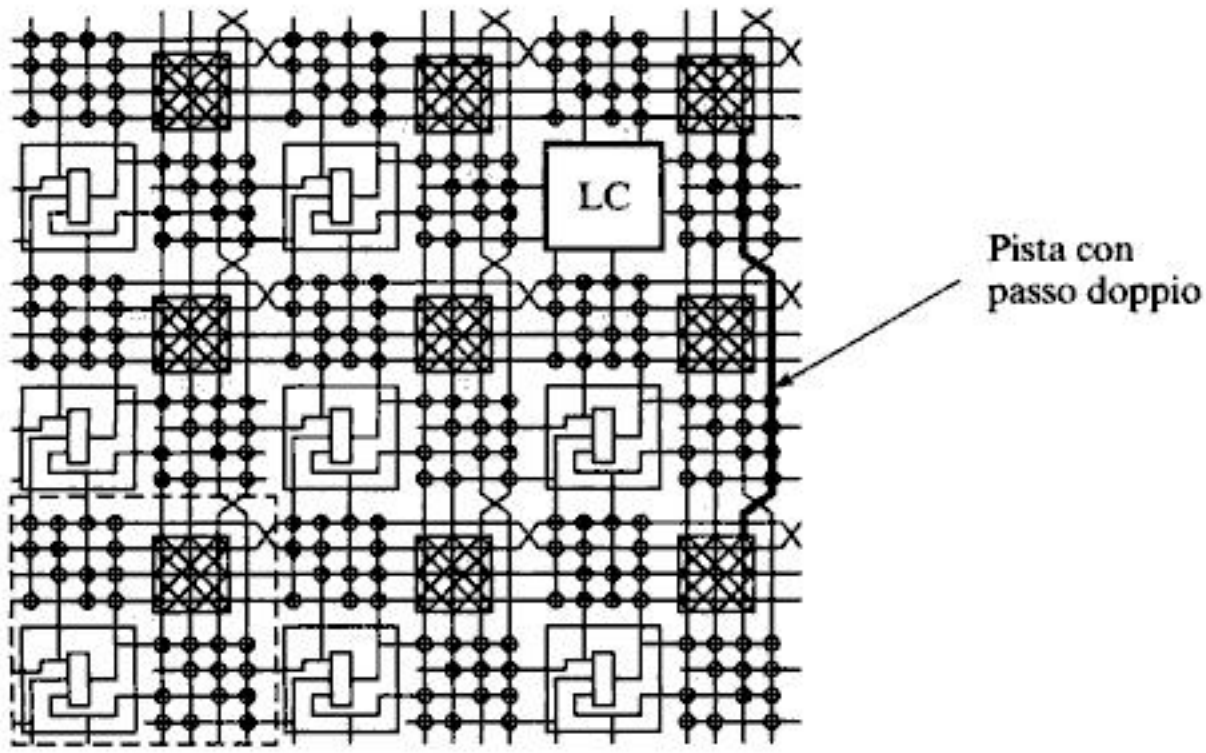


Figura 8-39 Architettura di interconnessione programmabile a rete con la sovrapposizione di una griglia 2 x 2 (*gentile concessione di Andre Dehon e John Wawrzyniek*).

L'architettura a rete rappresenta un mezzo flessibile e scalabile per connettere un gran numero di componenti; è piuttosto efficiente per le connessioni locali, poiché il numero di interruttori attraversati da un segnale è piccolo così come il fan-out. L'architettura a rete non si presta però bene alle interconnessioni globali: il ritardo causato dalla combinazione dei molti interruttori e dei notevoli carichi capacitivi può essere eccessivo. Molte architetture di FPGA basate sulle connessioni a rete offrono in realtà degli schemi alternativi per la realizzazione delle interconnessioni globali, come quello mostrato nella Figura 8-39. Oltre alle connessioni standard tra S-box, sono previste connessioni dirette anche tra S-box distanti due passi; l'eliminazione di una S-box lungo la pista di connessione ne riduce la resistenza totale. In modo analogo possiamo prevedere lunghe piste che connettono una S-box ogni 4, 8, o 16. Quello che in realtà stiamo creando è un certo numero di reti sovrapposte, ciascuna con diversa granularità (passo singolo, doppio, ecc.). I percorsi lunghi saranno quindi preferibilmente mappati sulle reti con il passo maggiore.

Materiale protetto da copyright

Mettere tutto insieme

Siamo ora in grado di assemblare una FPGA completa, unendo gli approcci visti per le celle logiche e per le interconnessioni. Si possono concepire (e lo sono state) molte architetture alternative. La decisione più importante da prendere all'inizio è il tipo di programmazione (scrivibile una volta, non volatile o volatile). Questo pone alcuni vincoli sul tipo di cella e di interconnessioni che possono essere usati. Dare una visione completa è al di là degli scopi di questo libro, quindi ci limitiamo a due architetture famose, che sono rappresentative. Il lettore interessato può trovare altre informazioni in [Trimberger94], [Smith97], [Betz99] e [George01].

La serie Altera MAX [Altera01] La famiglia MAX di dispositivi (Figura 8-40) appartiene alla classe delle FPGA non volatili (dette spesso EPLD, *Electrically Programmable Logic Devices*). Usa un modulo PAL (come quello presentato nella Figura 8-29) come modulo logico fondamentale. Il modulo LAB (*Logic Array Block*) varia poco tra i vari membri della famiglia: una grossa matrice AND programmabile, seguita da una matrice OR fissa e di dimensioni ridotte e un'inversione programmabile. Un LAB contiene tipicamente 16 macrocelle.

Le differenze maggiori stanno nelle architetture per le connessioni tra i vari LAB. I componenti di dimensioni più ridotte (MAX5000, MAX7000) usano un'architettura basata su matrice. La spina dorsale dei canali di connessione è costituita dalle uscite delle macrocelle con l'aggiunta dei segnali di ingresso del chip. Questi possono essere connessi in ingresso ai vari LAB attraverso dei punti di contatto programmabili. Il vantaggio di questa architettura, detta *PIA* (*Programmable Interconnect Array*), è che il ritardo dovuto alle connessioni tra blocchi è fisso e prevedibile (vedi Figura 8-41). Lo svantaggio è che non scala molto bene con le dimensioni del sistema. Questo è il motivo per cui i membri della serie di dimensioni maggiori (MAX9000) si devono affidare ad altri schemi. Con il numero di macrocelle che arriva a 560, l'approccio a canale unico diventa troppo lento non garantendo più prestazioni adeguate. In alternativa è stata usata un'architettura a rete, nella quale le singole macrocelle si possono connettere sia alle righe sia alle colonne, che sono piuttosto larghe (da 48 a 96 piste).

Gli EPLD mettono a disposizione fino a 15 000 porte logiche e tipicamente sono usati quando le prestazioni elevate sono irrinunciabili, mentre è preferibile ricorrere ad altre architetture quando è necessario implementare funzioni più complesse.

La serie Xilinx XC40xx Questa famosa famiglia di dispositivi programmabili via RAM combina l'approccio a lookup table per l'implementazione delle celle logiche e la struttura a rete per le interconnessioni. Il componente della serie di dimensioni maggiori (XC4085), contiene quasi 100 000 porte, usando una matrice 56×56 di CLB. L'architettura del blocco CLB è già stata mostrata nella Figura 8-34. Una caratteristica interessante è che il blocco CLB può anche essere configurato come una matrice di celle di memoria riscrivibili, usando la memoria delle lookup table nei blocchi F' e G'. A seconda della modalità selezionata, un singolo blocco CLB può essere configurato come una matrice da 16×2, 32×1 o 16×1 bit.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Questa caratteristica è molto utile, perché i moduli logici di grandi dimensioni hanno spesso bisogno di quantità altrettanto grandi di memoria.

Anche l'architettura delle interconnessioni è piuttosto ricca e combina una vasta gamma di risorse di interconnessioni, come mostrato nella Figura 8-42. Le reti sovrapposte consistono di segmenti di piste di lunghezza 1, 2 e 4. Alcuni componenti sono anche dotati di connesioni dirette che collegano blocchi CLB adiacenti, senza dover occupare le risorse per le connessioni globali. I segnali trasmessi attraverso le connessioni dirette subiscono un ritardo modesto, poiché il fan-out è piccolo. Queste connessioni *dirette* sono particolarmente funzionali per l'implementazione di blocchi aritmetici veloci, che richiedono molte connessioni locali critiche. Per implementare le interconnessioni globali, sono disponibili delle *linee lunghe*, che formano una griglia di segmenti di interconnessione metallici che corrono per l'intera lunghezza o larghezza della matrice. Questi sono dedicati a connessioni con fan-out elevato, a segnali critici dal punto di vista delle prestazioni o a connessioni distribuite per lunghe distanze (quali i bus). Sono inoltre disponibili delle piste speciali per la distribuzione dei clock.

Un argomento che abbiamo ignorato fino a ora riguardo le strutture di matrici programmabili è l'architettura di ingresso/uscita (I/O). Per una riusabilità massima, è importante che i terminali di I/O del componente siano configurabili e che mettano a disposizione una vasta gamma di opzioni in termini di direzionalità, livelli logici e capacità di pilotaggio. Un possibile stile per i blocchi di I/O (IOB), usato nella serie XC4000, è mostrato nella Figura 8-43: può essere programmato per funzionare come porta di ingresso, di uscita o bidirezionale. Comprende un flip-flop che può essere programmato per essere edge triggered o sensibile ai livelli del clock. Il controllo di slew-rate permette di modificare la capacità di pilotaggio e di ridurre i tempi di salita e discesa per segnali non critici.

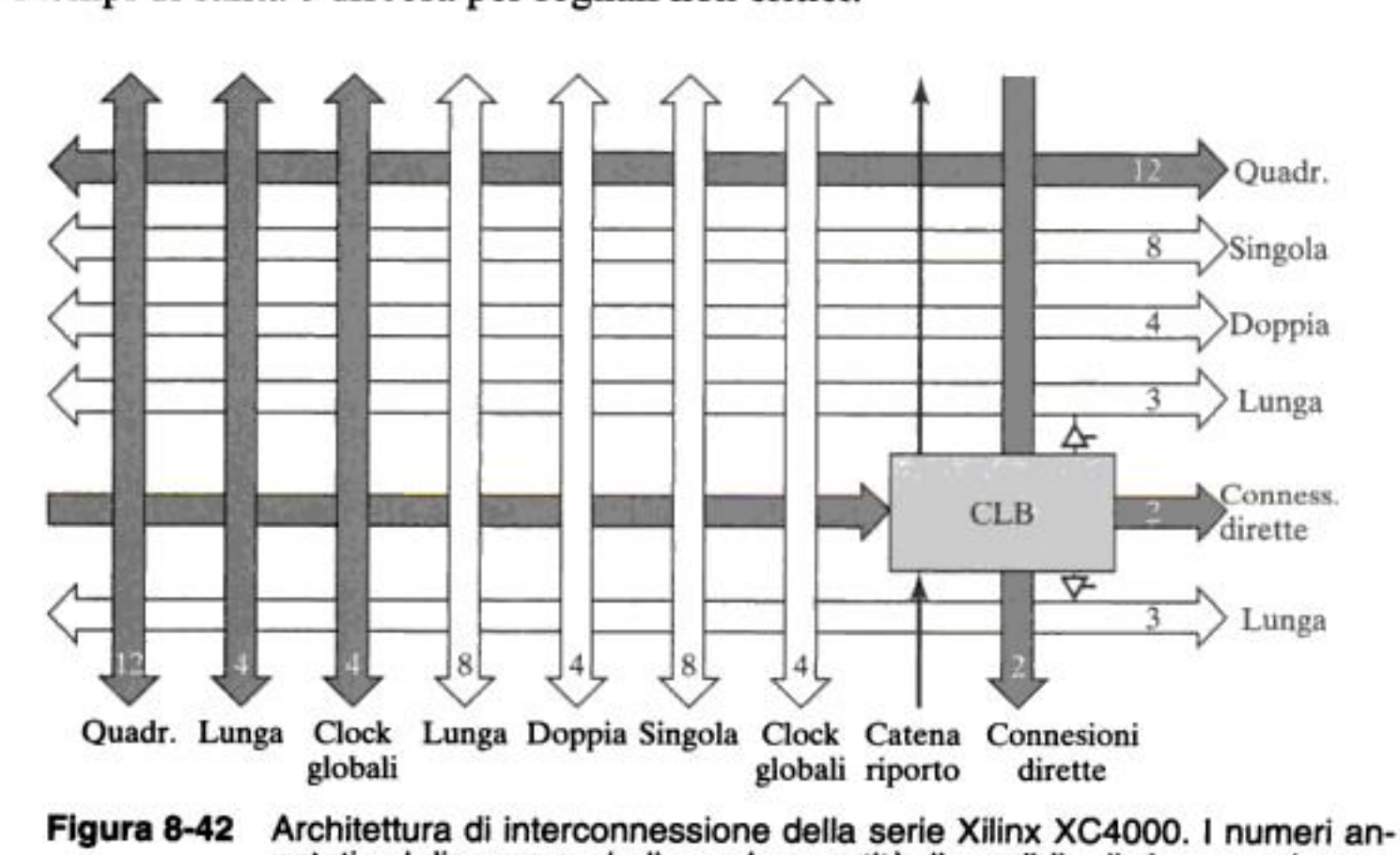


Figura 8-42 Architettura di interconnessione della serie Xilinx XC4000. I numeri annotati sul diagramma indicano la quantità disponibile di ciascuna risorsa.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

mentazioni efficienti e flessibili quando e dove è necessario. I sistemi integrati su singolo chip stanno diventando una combinazione di microprocessori integrati, banchi di memoria, DSP, acceleratori hardware in stile ASIC fisso, moduli parametrici e logica flessibile implementata con lo stile delle FPGA. Il bilanciamento tra questi componenti dipende dalle specifiche dell'applicazione e dalle richieste del mercato.

Esempio 8.12 Sistemi realizzati con piattaforme ibride

La Figura 8-45 mostra due piattaforme implementative opposte per applicazioni wireless. Il primo dispositivo, il Virtex-II Pro della Xilinx [XilinxVirtex01] è basato su una grossa matrice FPGA. Un microprocessore PowerPC, integrato al centro della matrice, è dedicato all'implementazione di funzionalità specifiche a livello di applicazione e di controllo a livello di sistema. Per raggiungere prestazioni più elevate per le applicazioni di elaborazione del segnale, è presente una matrice 18×18 di moltiplicatori. Questi componenti aggiuntivi permettono un significativo miglioramento in termini di prestazioni, potenza e area rispetto a un'implementazione a FPGA pura. Infine è disponibile un certo numero di transceivers molto veloci a 3.125-Gbps, per permettere la comunicazione seriale ad alta velocità con il mondo esterno.

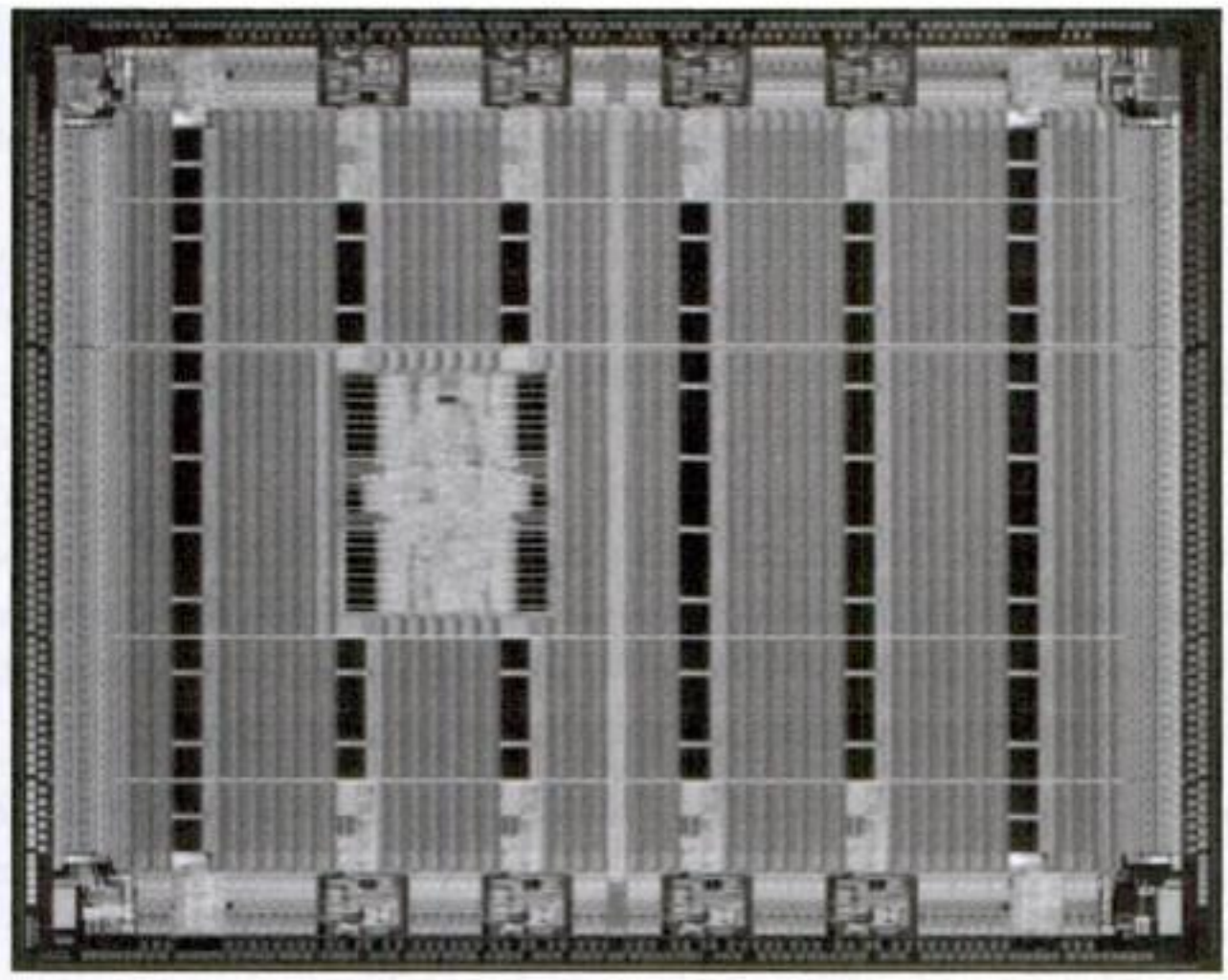
Un approccio in qualche modo opposto è rappresentato dal sistema della Figura 8-45b [Zhang00]. Il cuore di questo dispositivo è un microprocessore integrato ARM-7, che agisce come gestore globale del chip. Funzioni che richiedono alte prestazioni ed efficienza energetica sono demandate a matrici configurabili di unità funzionali quali moltiplicatori, ALU, memorie e generatori di indirizzo. Questi componenti possono essere combinati dinamicamente per formare processori per applicazioni specifiche. Il chip contiene anche una matrice a FPGA per funzioni che richiedono una granularità a livello di bit.

8.7 Riepilogo

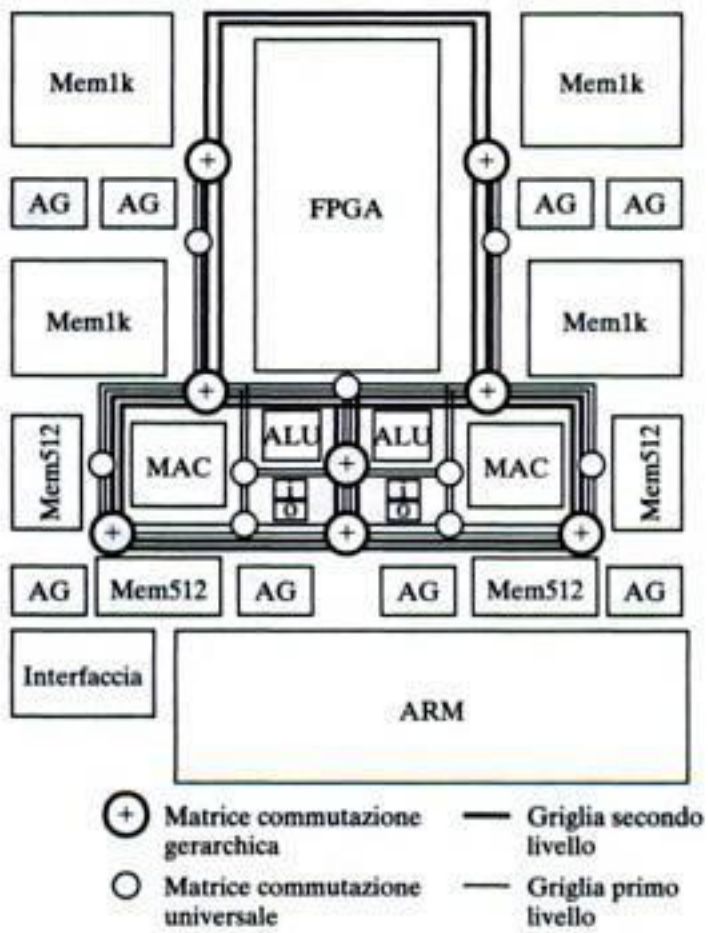
In questo capitolo, abbiamo scorso brevemente il complesso mondo delle tecnologie realizzative per i circuiti digitali. Nel corso degli ultimi decenni sono emerse rapidamente nuove strategie, mettendo a disposizione del progettista molte alternative. Queste tecniche progettuali e i relativi strumenti automatici stanno avendo un notevole impatto sul modo di progettare e rendono realizzabili sistemi incredibilmente complessi, come microprocessori e altri circuiti dedicati, a cui siamo ormai così abituati. In questo capitolo abbiamo esaminato i seguenti argomenti :

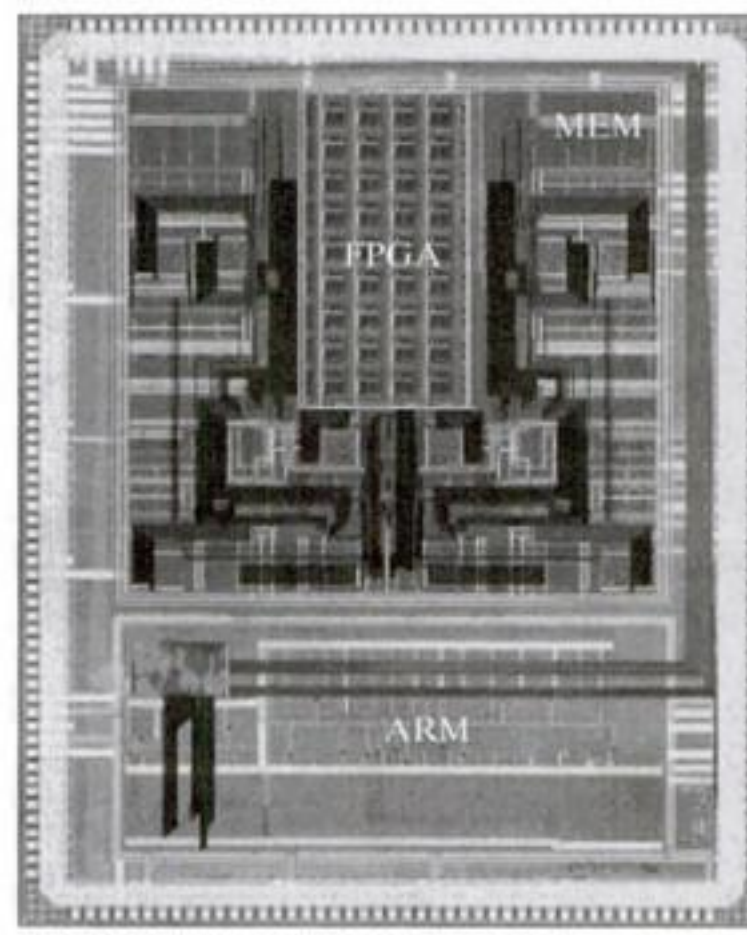
- La *progettazione dedicata*, in cui ciascun transistor è realizzato individualmente, è focalizzata all'ottimizzazione di area e prestazioni. Questo approccio è divenuto molto costoso e dovrebbe essere riservato alla progettazione di pochi moduli critici in cui siano irrinunciabili le prestazioni estreme o di celle di libreria istanziate molto spesso.

Materiale protetto da copyright



(a) La Xilinx Virtex-II Pro integra un microprocessore PowerPC in una FPGA (da Xilinx, Inc.).





(b) Il chip Maia combina microprocessore, acceleratori configurabili e FPGA [Zhang00].

Figura 8-45 Esempi di piattaforme di implementazione ibrida.

- L'approccio *semi-automatico*, basato su celle standard, è il cavallo di battaglia dell'industria della progettazione digitale odierna. Il vantaggio è l'alto grado di automazione; la sfida è la gestione dell'impatto delle tecnologie molto scalate.

Materiale protetto da copyright

- Per gestire la complessità crescente dei circuiti integrati, i progettisti si affidano sempre di più alla disponibilità di grosse *macrocelle* quali memorie, moltiplicatori e microprocessori. Questi moduli sono spesso forniti da terze parti e hanno incentivato un nuovo mercato basato sulla "*proprietà intellettuale*".
- Intraprendere un nuovo progetto per ogni nuova applicazione è divenuto troppo costoso. La maggior parte del mercato dei semiconduttori è ora concentrata su soluzioni flessibili che permettono di sfruttare lo stesso componente per diverse applicazioni, o attraverso la programmazione software o attraverso una riconfigurazione hardware. L'*hardware* configurabile permette di ritardare il momento in cui la funzione specifica viene fisicamente mappata sul circuito: abbiamo discusso diversi approcci in questo senso. Ritardare il momento in cui si realizza la funzione logica specifica comporta una penalità in termini di efficienza: maggiore è la flessibilità a disposizione, maggiori saranno le perdite in prestazioni e consumo.

Indubbiamente, in futuro assisteremo alla nascita di nuovi stili di progettazione. Acquisire familiarità con tutte le opzioni disponibili è sicuramente una parte essenziale dell'apprendimento del progettista digitale neofita. Speriamo che questo capitolo, sebbene non esaustivo, inviti il lettore a esplorare ulteriormente le varie possibilità. Un'osservazione conclusiva: nonostante il livello di automazione crescente della progettazione di circuiti digitali, nuove sfide si presentano in continuazione; sfide che richiedono una comprensione e un'intuizione profonde, che solo la mente umana può raggiungere.

8.8 Approfondimenti

La letteratura sulle metodologie e sull'automazione per la progettazione dei circuiti integrati digitali è esplosa negli ultimi decenni. Vale la pena di menzionare diversi contributi:

- Metodologie progettuali ASIC e FPGA: [Smith97]
- Architetture FPGA: [Trimberger94], [George01]
- Sistemi su singolo chip: [Chang99]
- Metodologie progettuali e tecnologia: [Bryant01]
- Sintesi circuitale: [DeMicheli94]

Gli sviluppi allo stato dell'arte nel dominio dell'automazione della progettazione sono in genere riportati in riviste quali *IEEE Transactions on CAD*, *IEEE Transactions on VLSI Systems*, e *IEEE Design and Test Magazine*. Le conferenze principali rivolte a questi argomenti sono: DAC (Design Automation Conference) e ICCAD (International Conference on CAD). Anche i siti web delle principali aziende di automazione della progettazione elettronica (Cadence, Synopsys, Mentor ecc.) forniscono molte informazioni preziose.

Riferimenti scientifici

[Altera01] Altera Device Index, *http://www.altera.com/products/devices/dev-index.html*, 2001.

[Avanti01] Saturn Efficient and Concurrent Logical and Physical Optimization of SoC Timing, Area and Power, *http://www.synopsis.com/product/avmrg/saturn_ds.html*.

Materiale protetto da copyright

[Betz99] V. Betz, J. Rose, and A. Marquardt, *Architecture and CAD for Deep-Submicron FPGAs*, Kluwer International Series in Engineering and Computer Science, Kluwer Academic Publishers, 1999.

[Brayton84] R. Brayton et al., *Logic Minimization Algorithms for VLSI Synthesis*, Kluwer Academic Publishers, 1984.

[Bryant01] R. Bryant, T. Cheng, A. Kahng, K. Keutzer, W. Maly, R. Newton, L. Pileggi, J. Rabaey, and A. Sangiovanni-Vincentelli, "Limitations and Challenges of Computer-Aided Design Technology for CMOS VLSI," *IEEE Proceedings*, pp. 341–365, March 2001.

[Cadabra01] AbraCAD Automated Layout Creation, *http://www.cadabratech.com/?id=145products*, Cadabra Design Automation.

[Chang99] H. Chang et al., "Surviving the SOC Revolution: A Guide to Platform-Based Design," Kluwer Academic Publishers, 1999.

[DeMicheli94] G. De Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, 1994.

[El-Ayat89] K. El-Ayat, "A CMOS Electrically Configurable Gate Array," *IEEE Journal of Solid State Circuits*, vol. SC-24, no. 3, pp. 752–762, June 1989.

[George01] V. George and J. Rabaey, *Low-Energy FPGAs*, Kluwer Academic Publishers, 2001.

[Grundman97] W. Grundmann, D. Dobberpuhl, R. Allmon, and N. Rethman "Designing High-Performance CMOS processores Using Full Custom Techniques," *Proceedings Design Automation Conference*, pp. 722–727, Anaheim, June 1997.

[Hill85] D. Hill, "S2C—A Hybrid Automatic Layout System," *Proc. ICCAD-85*, pp. 172–174, November 1985.

[ModuleCompiler01] Synopsys Module Compiler Datasheet, *http://www.synopsys.com/products/datapath/datapath.html*, Synopsys, Inc.

[Philips99] The Nexperia System Silicon Implementation Platform, *http://www.semiconductors.philips.com/platforms/nexperia/*, Philips Semiconductors.

[Pileggi02] Pileggi, Schmit et al., "*Via Patterned Gate Array*," CMU Center for Silicon System Implementation Technical Report Series, no. CSSI 02-15, April 2002.

[Prolific01] The ProGenensis Cell Compiler, *http://www.prolificinc.com/progenesis.html*, Prolific, Inc.

[Rabaey00] J. Rabaey, "Low-Power Silicon Architectures for Wireless Applications," *Proceedings ASPDAC Conference*, Yokohama, January 2000.

[Silva01] J. L. da Silva Jr., J. Shamberger, M. J. Ammer, C. Guo, S. Li, R. Shah, T. Tuan, M. Sheets, J. M. Rabaey, B. Nikolic, A. Sangiovanni-Vincentelli, P. Wright, "Design Methodology for PicoRadio Networks," *Proc. DATE Conference*, Munich, March 2000.

[Smith97] M. Smith, *Application-Specific Integrated Circuits*, Addison-Wesley, 1997.

[Sylvester98] D. Sylvester and K. Keutzer, "Getting to the Bottom of Deep Submicron," *Proc. ICCAD Conference*, pp. 203, San Jose, November 1998.

[Trimberger94] S. Trimberger, *Field-Programmable Gate Array Technology*, Kluwer Academic Publishers, 1994.

[Veendrick92] H. Veendrick, *MOS IC's: From Basics to ASICS*, Wiley-VCH, 1992.

[Xilinx4000] The Xilinx-4000 Product Series, *http://www.xilinx.com/apps/4000.htm*, Xilinx, Inc.

[XilinxVirtex01] Virtex-II Pro Platform FPGAs, *http://www.xilinx.com/xlnx/xil_prodcat_landing page.jsp?title=Virtex-II+Pro+FPGAs*, Xilinx, Inc.

[Xtensa01] Xtensa Configurable Embedded processore Core, *http://www.tensilica.com/technology.html*, Tensilica.

[Zhang00] H. Zhang, V. Prabhu, V. George, M. Wan, M. Benes, A. Abnous, and J. Rabaey, "A 1 V Heterogeneous Reconfigurable processore IC for Baseband Wireless Applications," *Proc. ISSCC*, pp. 68–69, February 2000.

Esercizi

Per problemi ed esercizi sulle metodologie progettuali, fate riferimento a **http://bwrc.eecs.berkeley.edu/IcBook**.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Caratterizzazione di celle logiche

Sfortunatamente non è mai stato adottato alcun modello standard per i ritardi nelle celle standard. Ciascun produttore ha i propri metodi per la caratterizzazione delle celle. Perfino nell'ambito di un singolo strumento, sono spesso disponibili diversi modelli dei ritardi, che fanno compromessi diversi tra precisione e velocità. I concetti fondamentali sono però piuttosto simili e sono strettamente in relazione con quelli introdotti nei Capitoli 5 e 6. In questo paragrafo scegliamo quindi di concentrarci su un singolo insieme di modelli, più precisamente quelli usati in Design Compiler della Synopsys [DesignCompiler01], uno degli strumenti di sintesi più comuni. Una volta scelto un modello, questo deve essere usato per tutte le celle; in altre parole non può essere cambiato da cella a cella.

Il ritardo totale è dato da quattro componenti, come si vede nella Figura E-1:

$$D_{tot} = D_I + D_T + D_S + D_C. \tag{E.1}$$

D_I rappresenta il *ritardo intrinseco*, che è il ritardo senza alcun carico in uscita. D_T è la componente di transizione, ovvero quella parte di ritardo causata dal carico in uscita. D_S è la parte di ritardo dovuta alla *pendenza dell'ingresso*. Infine D_C è il *ritardo della connessione* che segue la porta logica. Tutti i ritardi sono caratterizzati sia per la transizione di salita sia per quella di discesa.

Il modello più semplice per il ritardo di transizione è il modello lineare del Capitolo 5. Vale

$$D_T = R_{driver}(\Sigma C_{porta} + C_{pista}), \tag{E.2}$$

dove ΣC_{porta} è la somma delle capacità di tutti i pin di ingresso delle porte connesse all'uscita della porta logica che stiamo caratterizzando e C_{pista} è la stima della capacità delle piste di interconnessione. Il ritardo di pendenza D_S è approssimabile con una funzione lineare del ritardo di transizione D_T della porta precedente, scritto come

$$D_S = S_S\, D_{Tprec} \tag{E.3}$$

dove S_S è il *fattore di sensibilità alla pendenza* e D_{Tprec} è il ritardo di transizione dello stadio precedente.

La caratterizzazione di una libreria di celle deve quindi fornire i seguenti parametri, per entrambe le transizioni di salita e discesa e rispetto a ciascuno dei terminali di ingresso:

- ritardo intrinseco;
- capacità del terminale di ingresso;
- resistenza equivalente di uscita;
- sensibilità alla pendenza.

In aggiunta ai modelli della cella, lo strumento di sintesi deve anche avere accesso a un modello per le interconnessioni. Dato che la lunghezza delle piste à ignota prima del piazzamento delle celle, le stime di C_{pista} e R_{pista} sono ricavate sulla base della dimensione del blocco e del fan-out della porta. La lunghezza della pista è spesso proporzionale al numero di porte a cui si deve connettere.

Materiale protetto da copyright

ridurre l'impegno di memoria e il numero di misure di caratterizzazione, viene considerata una matrice limitata di carichi esterni e valori di pendenza del segnale di ingresso; si usa poi un'interpolazione lineare per determinare i valori mancanti.

Esempio E.2 Modello di ritardo con lookup table

Di seguito è riportata una caratterizzazione (parziale) di una cella AND a due ingressi (AND2), progettata in una tecnologia CMOS da 0.25 μm (*per gentile concessione di ST Microelectronics*). I ritardi sono stati misurati con capacità di uscita di 7 fF, 35 fF, 70 fF e 140 fF e pendenze del segnale di ingresso di 40 ps, 200 ps, 800 ps e 1.6 ns.

```
cell(AND) {
 area : 36 ;
 pin(Z) {
  direction : output ;
  function : "A*B";
  max_capacitance : 0.14000 ;

  timing() {
   related_pin : "A" ; /* ritardo tra il terminale di ingresso A e di uscita Z */
cell_rise {
     values( "0.10810, 0.17304, 0.24763, 0.39554", \
           "0.14881, 0.21326, 0.28778, 0.43607", \
           "0.25149, 0.31643, 0.39060, 0.53805", \
           "0.35255, 0.42044, 0.49596, 0.64469" ); }
    rise_transition {
     values( "0.08068, 0.23844, 0.43925, 0.84497", \
           "0.08447, 0.24008, 0.43926, 0.84814", \
          "0.10291, 0.25230, 0.44753, 0.85182", \
           "0.12614, 0.27258, 0.46551, 0.86338" );}
  cell_fall(table_1) {
   values( "0.11655, 0.18476, 0.26212, 0.41496", \
           "0.15270, 0.22015, 0.29735, 0.45039", \
           "0.25893, 0.32845, 0.40535, 0.55701", \
           "0.36788, 0.44198, 0.52075, 0.67283" );}
  fall_transition(table_1) {
   values( "0.06850, 0.18148, 0.32692, 0.62442", \
           "0.07183, 0.18247, 0.32693, 0.62443", \
           "0.09608, 0.19935, 0.33744, 0.62677", \
           "0.12424, 0.22408, 0.35705, 0.63818" );}
  intrinsic_rise : 0.13305 ; /* ritardi senza carico esterno*/
  intrinsic_fall : 0.13536 ;
 }
```

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Sintesi del circuito

Sintesi circuitale, logica e architetturale

Una delle proposte più intriganti che si possono fare a un progettista che deve generare in poco tempo un circuito con specifiche stringenti è quella di uno strumento che traduca automaticamente le sue specifiche in un circuito funzionante che rispetta tutti i vincoli. Una delle ragioni principali per cui i circuiti a semiconduttore hanno raggiunto l'attuale complessità è l'esistenza di questi strumenti di sintesi, almeno entro un certo limite. La sintesi può essere definita come la trasformazione tra due differenti viste del sistema. Tipicamente si tratta di una traduzione da una specifica *comportamentale* di un'entità circuitale in una descrizione *strutturale*. In parole povere, si traduce la descrizione della funzione che un modulo deve svolgere (il suo comportamento) in una composizione, cioè un'interconnessione di elementi (la sua struttura). La sintesi può essere definita a tutti i livelli di astrazione: circuitale, logico e architetturale. Nella Figura F-1 è riportata una panoramica dei vari livelli di sintesi e del relativo risultato. La procedura di sintesi può anche essere differente a seconda dello stile di implementazione che si è deciso si usare. La sintesi logica, per esempio, traduce una descrizione logica data da un insieme di equazioni booleane in una rete di porte logiche. Le tecniche coinvolte in questa fase dipendono fortemente dalla scelta fra uno stile implementativo a due livelli (PLA) o a livelli multipli (celle standard o gate array). Descriveremo brevemente le procedure di sintesi per ciascuno dei diversi livelli di modellazione. Per maggiori informazioni e una visione più approfondita della sintesi di un sistema si può fare riferimento a [DeMicheli94].

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Tabella F-1 Tabella di verità minimizzata per l'implementazione PLA del full-adder. I trattini (–) indicano che l'ingresso corrispondente non è incluso nel termine prodotto.

A	*B*	C_I	*S*	C_o
1	1	1	1	–
0	0	1	1	–
0	1	0	1	–
1	0	0	1	–
1	1	–	–	1
1	–	1	–	1
–	1	1	–	1

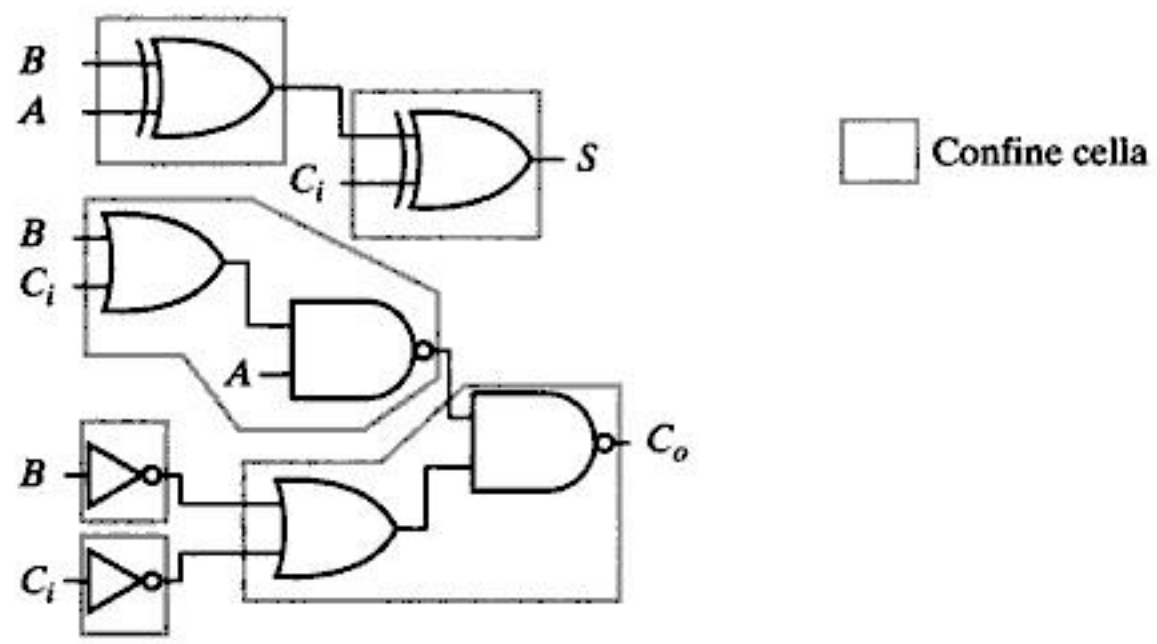


Figura F-2 Implementazione a celle standard di un full-adder, secondo quanto generato dalla sintesi a livelli logici multipli.

ridotto riordinando i segnali. Questo richiede che il progettista riconosca questo percorso come il più critico, fatto non ovvio da una semplice osservazione delle equazione del full-adder.

Sintesi architetturale

Nel campo della sintesi, quella architetturale è l'ultimo progresso. È anche indicata con il termine di *sintesi comportamentale* o *ad alto livello*. Il suo scopo è di generare una vista strutturale di un'architettura, data una descrizione comportamentale della funzione da svolgere e un insieme di vincoli in termini di prestazioni, area e/o potenza. Ciò corrisponde a individuare quali risorse architetturali sono necessarie per svolgere il compito (unità di elaborazione, memorie, bus e controllori), associando le operazioni comportamentali a risorse hardware e

Materiale protetto da copyright

determinando l'ordine di esecuzione delle operazioni nell'architettura generata. Nel gergo della sintesi queste funzioni sono dette *allocazione (allocation), assegnazione (assignment)* e *organizzazione (scheduling)* [Gajski92, DeMicheli94]. Sebbene queste operazioni rappresentino il nucleo della sintesi architetturale, vi sono altri passi che possono avere conseguenze drastiche sulla qualità della soluzione generata. Per esempio, delle trasformazioni ottimizzate che manipolino la descrizione comportamentale di partenza, possono portare a una soluzione finale superiore in termini di area o velocità. L'architettura *pipeline* è un esempio tipico di tali trasformazioni. In un certo senso questa fase del processo di sintesi è analoga alle trasformazioni per l'ottimizzazione nei compilatori software.

Esempio F.2 Sintesi architetturale

Per illustrare il concetto e le potenzialità della sintesi architetturale, consideriamo il semplice diagramma di flusso computazionale della Figura F-3: descrive un programma che riceve in ingresso tre numeri *a, b* e *c* e produce in uscita la loro somma *x*.

Nella Figura F-4 ne sono mostrate due possibili implementazioni, secondo quanto generato dal sistema di sintesi HYPER [Rabaey91]. La prima istanza richiede quattro cicli di clock e condivide nel tempo il bus di ingresso così come il sommatore. La seconda architettura realizza il programma in un singolo ciclo di clock. Per ottenere questo risultato è stato necessario partizionare in una pipeline l'algoritmo; cioè si sovrappongono più iterazioni del calcolo. La maggiore velocità si trasforma naturalmente in maggiori costi dell'hardware (un sommatore in più) registri aggiuntivi e un'architettura del bus maggiormente dedicata, compresa la necessità di tre porte di ingresso. Entrambe le architetture sono state generate automaticamente, data una descrizione comportamentale e il vincolo sui cicli di clock, che include la trasformazione in architettura pipeline.

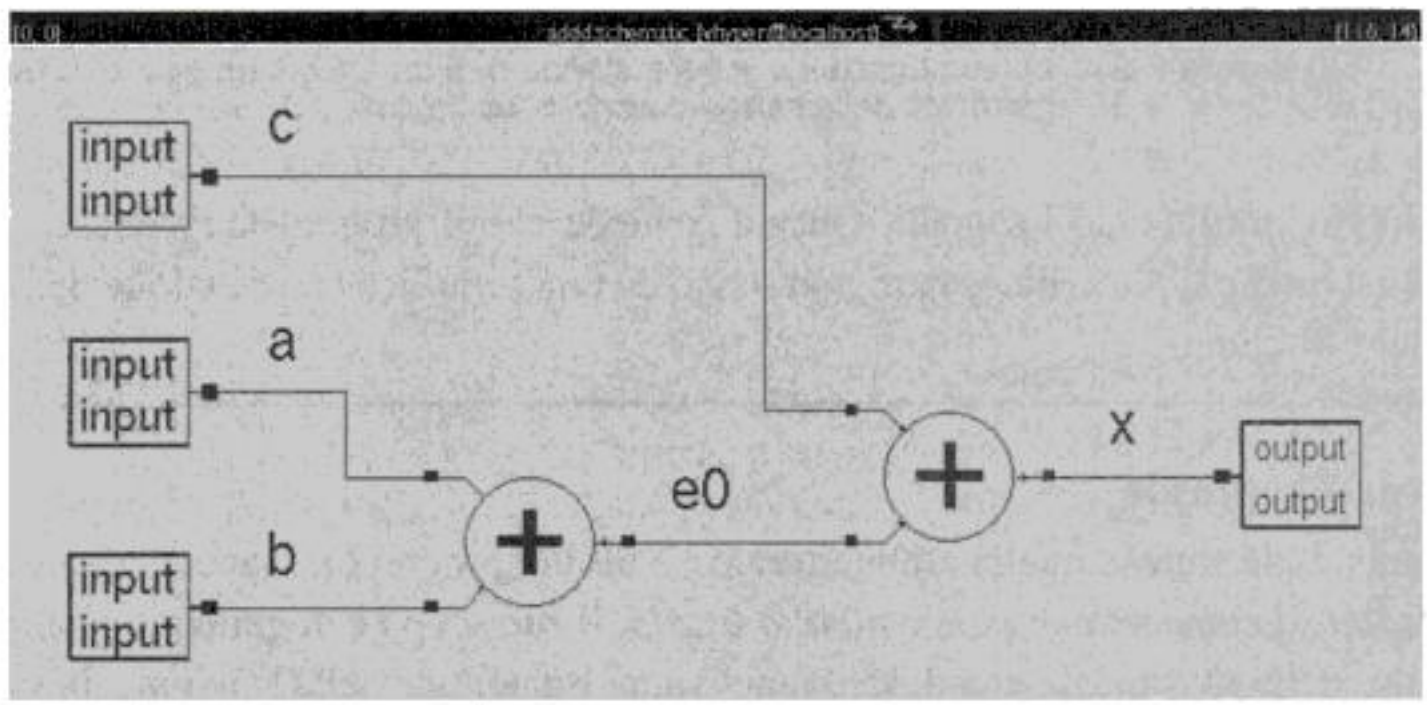
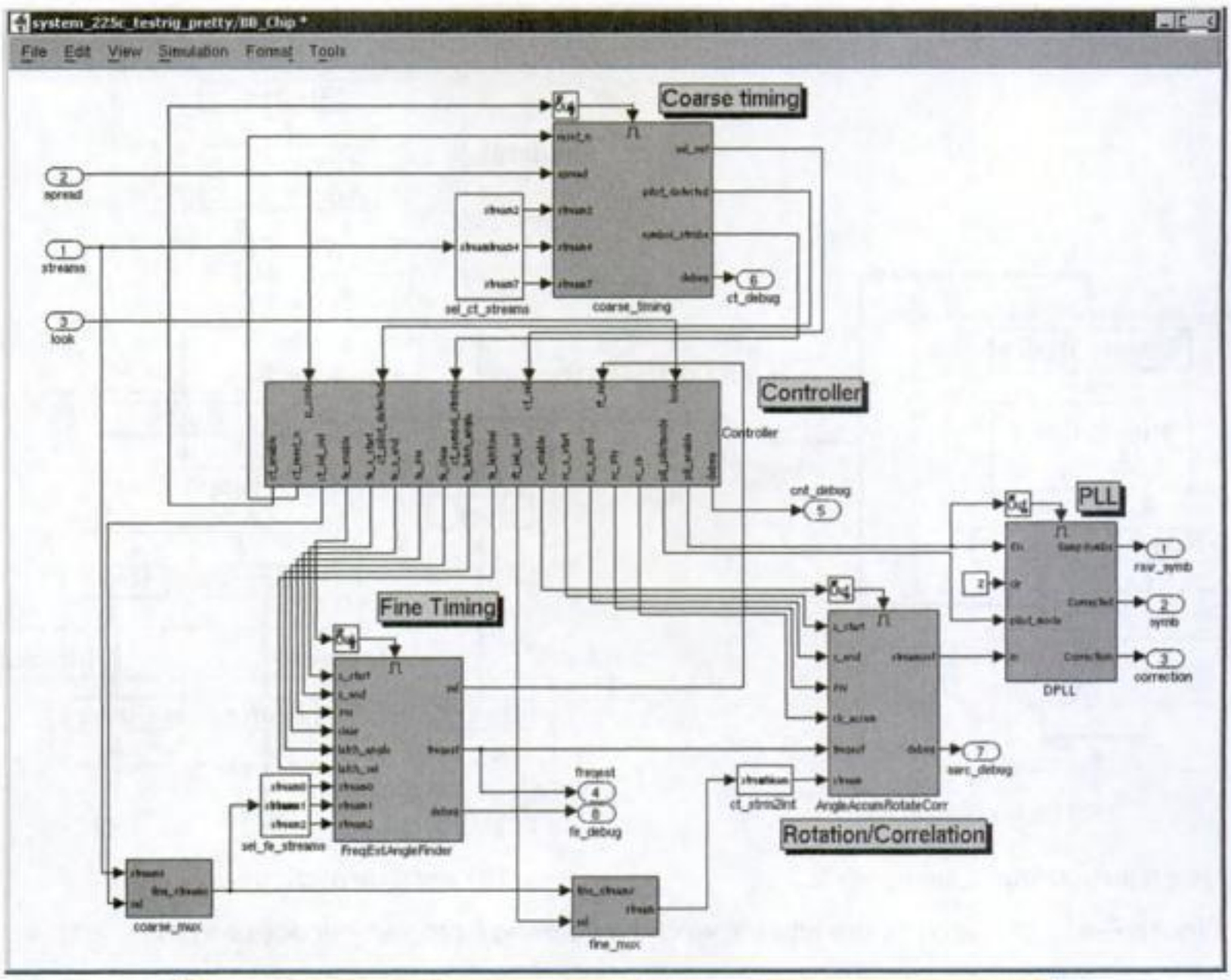


Figura F-3 Semplice programma che realizza la somma di tre numeri.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



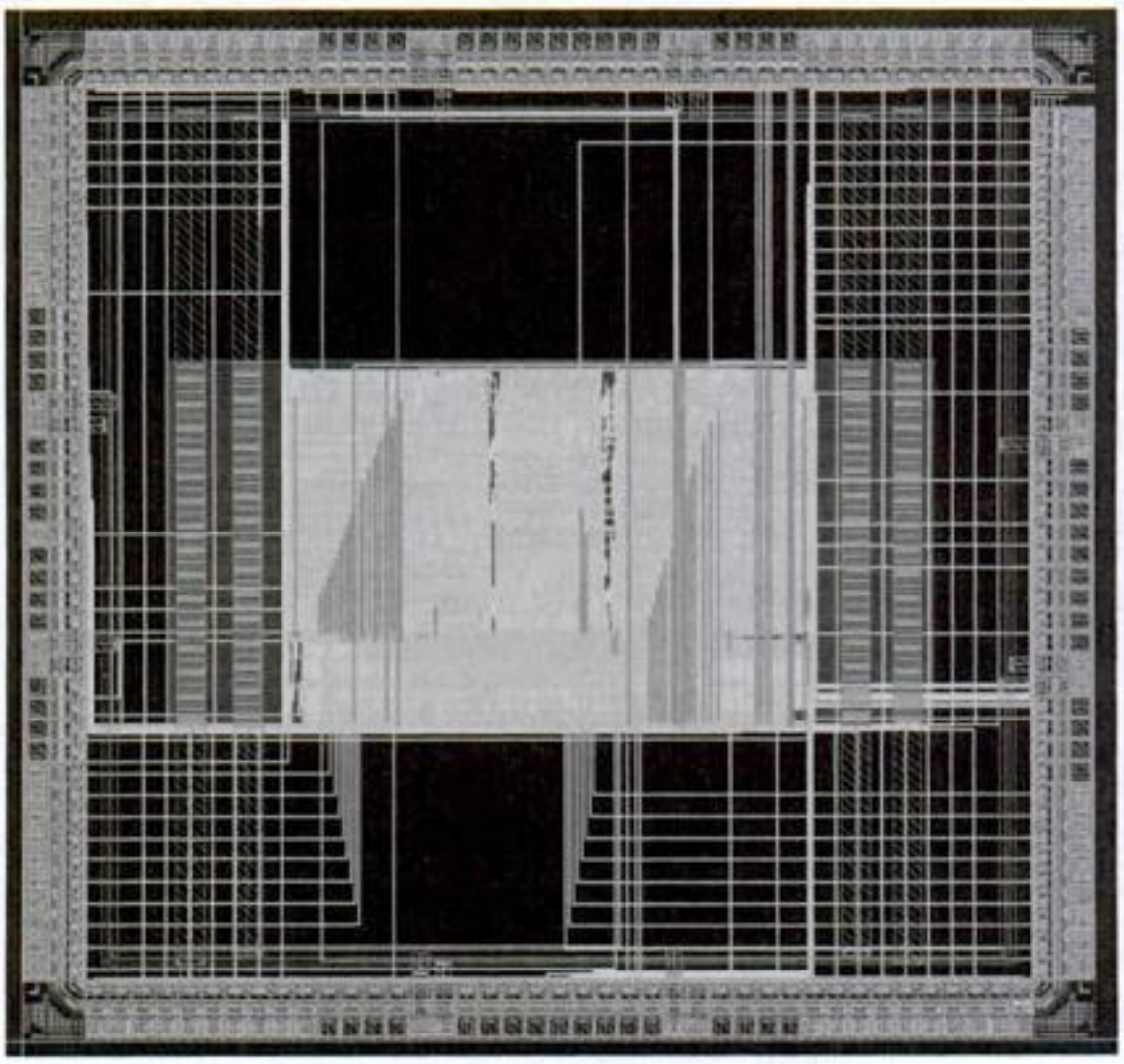


Figura F-5 Sintesi architetturale di un processore di banda base per applicazioni wireless da Simulink (a) al silicio (b). L'area del nucleo del chip, che è limitata dai pad, misura solamente 2 mm^2 in una tecnologia CMOS da 0.18 μm, e comprende 600 000 transistor. La densità elevata (0.3 transistor/μm^2) dimostra l'efficacia degli strumenti di progettazione fisica odierna.

Materiale protetto da copyright

[Chu86] K. Chu and D. Pulfrey, "Design Procedures for Differential Cascode Logic," *IEEE Journal of Solid State Circuits*, vol. SC-21, no. 6, Dec. 1986, pp. 1082–1087.

[Davis01] W.R. Davis, N. Zhang, K. Camera, F. Chen, D. Markovic, N. Chan, B. Nikolic, R.W. Brodersen, "A Design Environment for High Throughput, Low Power, Dedicated Signal Processing Systems," *Proceedings CICC 2001*, San Diego, 2001.

[DesignCompiler] Design Compiler Technical Datasheet, *http://www.synopsys.com/products/logic/design_compiler.html*, Synopsys, Inc.

[DeMan86] H. De Man, J. Rabaey, P. Six, and L. Claesen, "Cathedral-II: A Silicon Compiler for Digital Signal Processing," *IEEE Design and Test*, vol. 3, no. 6, pp. 13–25, December 1986.

[DeMicheli94] G. De Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, 1994.

[Fishburn85] J. Fishburn and A. Dunlop, "TILOS: A Polynomial Programming Approach to Transistor Sizing," *Proceedings ICCAD-85*, pp. 326–328, Santa Clara, 1985.

[Gajski92] D. Gajski, N. Dutt, A. Wu, and S. Lin, *High-Level Synthesis—Introduction to Chip and System Design*, Kluwer Academic Publishers, 1992.

[Kurup97] P. Kurub and T. Abassi, *Logic Synthesis using Synopsys*, Kluwer Academic Publishers, 1997.

[Mathworks01] Matlab and Simulink, *http://www.mathworks.com*, The Mathworks

[Northrop01] G. Northrop, P. Lu, "A Semicustom Design Flow in High-Performance Microprocessor Design," *Proceedings 38th Design Automation Conference*, Las Vegas, June 2001.

[Rabaey91] J. Rabaey, C. Chu, P. Hoang and M. Potkonjak, "Fast Prototyping of Datapath-Intensive Architectures," *IEEE Design and Test*, vol. 8, pp. 40–51, 1991.

[Silva01] J.L. da Silva Jr., J. Shamberger, M.J. Ammer, C. Guo, S. Li, R. Shah, T. Tuan, M. Sheets, J.M. Rabaey, B. Nikolic, A. Sangiovanni-Vincentelli, P. Wright, "Design Methodology for PicoRadio Networks," *Proceedings DATE Conference*, Munich, March 2000.

[SystemGenerator] The Xilinx System Generator for DSP, *http://www.xilinx.com/xlnx/xil_prodcat_product.jsp?title=system_generator*, Xilinx, Inc.

Materiale protetto da copyright

CAPITOLO

9

Interconnessioni e parassiti

Pilotare grandi capacità

Comportamento da linea di trasmissione delle piste

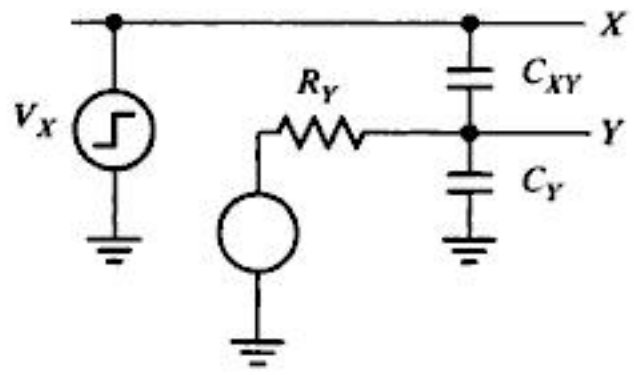
Integrità del segnale in presenza dei parassiti delle interconnessioni

Rumore nelle linee di alimentazione

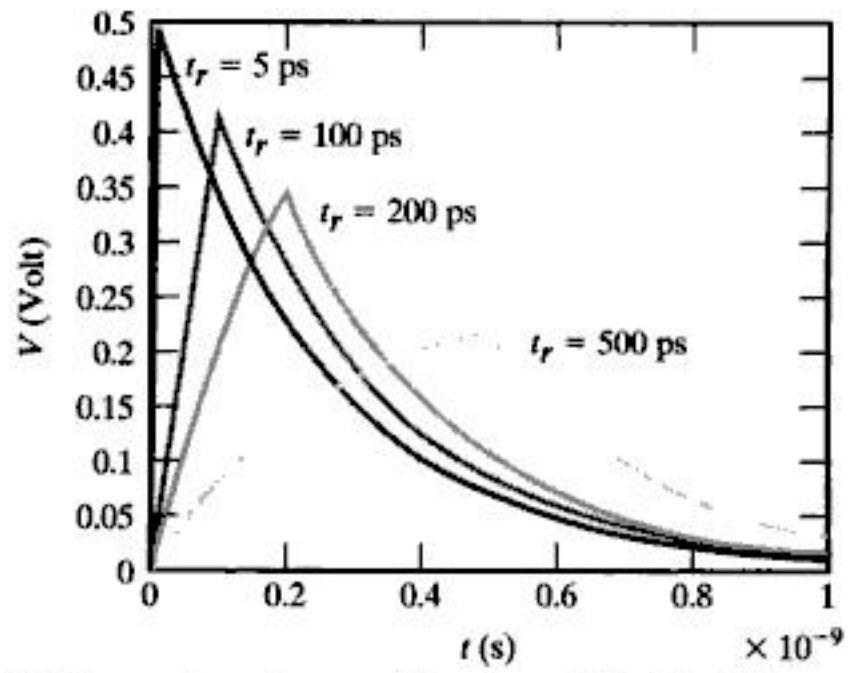
9.1 Introduzione
9.2 Capacità parassite
  9.2.1 Capacità e robustezza: cross talk
  9.2.2 Capacità e prestazioni dei circuiti CMOS
9.3 Resistenze parassite
  9.3.1 Resistenza e robustezza: cadute di tensione
  9.3.2 Elettromigrazione
  9.3.3 Resistenza e prestazioni: ritardo RC
9.4 Induttanze parassite*
  9.4.1 Induttanze e robustezza: cadute di tensione
  9.4.2 Induttanze e prestazioni: linee di trasmissione
9.5 Tecniche di interconnessione avanzate
  9.5.1 Circuiti a escursione logica ridotta
  9.5.2 Tecniche di trasmissione nel dominio della corrente
9.6 Prospettive: reti su singolo chip
9.7 Riepilogo
9.8 Approfondimenti

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Linea pilotata Y com l'interferente X



(b) Risposta in tensione per diversi tempi di salita di V_x (da 0 a 2.5 V) (R_x = 10 kΩ, C_y = 20 fF, C_{xy} = 5 fF)

Figura 9-3 Accoppiamento capacitivo a una linea pilotata.

Linee pilotate

Se la linea Y è statica, cioè pilotata attraverso una resistenza R_Y, un gradino sulla linea X causa un fenomeno transitorio sulla linea Y (vedi Figura 9-3a). Il transitorio decade con una costante di tempo $\tau_{XY} = R_Y(C_{XY} + C_Y)$. L'impatto effettivo sulla "linea vittima" dipende fortemente dal tempo di salita (discesa) del segnale interferente. Se il tempo di salita è confrontabile o maggiore della costante di tempo, il valore di picco del disturbo è ridotto. Questo è evidenziato dai risultati di simulazione mostrati nella Figura 9-3b. Ovviamente mantenere bassa l'impedenza del dispositivo che pilota la pista (e quindi τ_{XY}) contribuisce a ridurre le conseguenza del cross talk capacitivo. Il transistor *bleeder*, aggiunto alla porta dinamica o alla pista precaricata, rappresenta un eccellente esempio di come la riduzione dell'impedenza contribuisca a controllare il rumore.

Per riassumere, le conseguenze del cross talk sull'integrità del segnale di nodi statici sono piuttosto limitate. Gli impulsi che ne conseguono possono comunque causare dei malfunzionamenti in elementi sequenziali connessi alla pista e dovrebbero quindi essere controllati con cura. L'effetto più importante è però un aumento dei ritardi, che prenderemo in esame più avanti.

Tecniche di progettazione: ridurre gli effetti del cross talk capacitivo

Il cross talk è una sorgente di rumore proporzionale al segnale. Questo significa che scalare i livelli di segnale per aumentare i margini di rumore non è di aiuto, perché le sorgenti di rumore scalano dello stesso fattore. L'unica possibilità per affrontare il problema consiste nel tenere sotto controllo la geometria del circuito oppure di adottare delle convenzioni per la rappresentazione dei segnali che siano meno sensibili agli accoppiamenti. Possiamo definire le seguenti regole fondamentali, secondo quanto sostenuto in [Dally98]:

1. Se possibile, evitare del tutto nodi flottanti. Nodi sensibili a problemi di cross-talk, quali bus a precarica, dovrebbero essere dotati di circuiti di rigenerazione del segnale per ridurre l'impedenza.

Materiale protetto da copyright

dentemente. Ipotizziamo che tutti gli N ingressi commutino nello stesso istante. A causa delle capacità mutue, il ritardo della pista *k-esima* è funzione della commutazione delle piste vicine, $k - 1$ e $k + 1$. In base alla formula di Elmore, ricaviamo una buona approssimazione di questo ritardo:

$$t_{p,k} = gC_W(0{,}38R_W + 0{,}69R_D) \tag{9.3}$$

dove $C_W = c_wL$ e $R_W = r_wL$; c_w e r_w rappresentano rispettivamente la capacità verso massa e la resistenza della pista, mentre R_D è la resistenza equivalente della sorgente che pilota le piste. Il fattore correttivo g tiene conto dell'effetto del cross talk ed è funzione del rapporto $r = c_i/c_w$ e dell'attività sulla pista; c_i rappresenta la capacità mutua tra le piste per unità di lunghezza. Il valore di g per una serie di situazioni rappresentative è riportato nella Tabella 9-1. Quando tutte le tre piste ($k - 1$, k, $k + 1$) commutano nello stesso senso, la capacità mutua non entra in gioco e $g = 1$. Il caso peggiore si verifica quando entrambe le linee vicine commutano in senso opposto rispetto alla pista k, portando $g = 1 + 4r$. Per il caso plausibile in cui $c_i = c_w$, $g = 5$. Quindi il ritardo della pista può variare del 500% tra il caso migliore e quello peggiore, semplicemente in funzoine del verso della commutazione dei segnali.

Tabella 9-1 Fattore di ritardo g per la pista del bus in funzione delle ommutazioni simultanee sulle linee vicine. La tabella contiene solo alcuni esempi rappresentativi ma può essere facilmente estesa per contenere un insieme completo di tutte le possibili combinazioni delle commutazioni. I simboli —, ↑ e ↓ rappresentano rispet-tivamente commutazioni inesistente, positiva e negativa.

bit $k-1$	bit k	bit $k+1$	Fattore ritardo g
↑	↑	↑	1
↑	↑	—	$1 + r$
↑	↑	↓	$1 + 2r$
—	↑	—	$1 + 2r$
—	↑	↓	$1 + 3r$
↓	↑	↓	$1 + 4r$

Tecniche di progettazione: fabbricazione di circuiti con ritardi delle piste prevedibili

pre affrontare il problema del cross talk, che rende il ritardo della pista sempre più imprevedibile, il progettista ha a disposizione alcune tecniche, tra cui:

1. **Valutare e migliorare**. Dopo simulazioni e valutazioni accurate, si individuano i colli di bottiglia nei ritardi e si modifica di conseguenza il circuito.

Materiale protetto da copyright

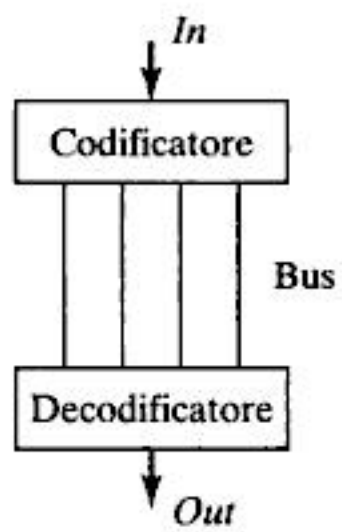


Figura 9-7 Codificare i dati per eliminare le configurazioni del caso peggiore può rendere un bus più veloce.

di codifica e decodifica (vedi Figura 9-7). Sebbene queste implichino maggiore hardware e un aumento dei ritardi, possono comunque essere di aiuto per ridurre il ritardo totale fino a un fattore 2 per bus grandi [Sotiriadis01]. La codifica dei dati prima della trasmissione lungo un bus può anche aiutare a ridurre il consumo di potenza minimizzando il numero di commutazioni [Stan95]. Però quest'ultimo approccio non prende in considerazione di per se stesso il cross talk. ■

Carico capacitivo e prestazioni

L'aumento del valore delle capacità delle interconnessioni, specialmente di quelle globali, mette in evidenza la necessità di disporre di circuiti in grado di pilotare le interconnessioni in modo efficiente, caricandone (e scaricandone) le capacità a velocità sufficiente. Tale necessità e resa più critica dal fatto che nei sistemi complessi una singola porta deve pilotare fan-out elevati e quindi ha un carico capacitivo elevato. Alcuni esempi tipici di piste con carico capacitivo elevato all'interno di un chip sono i bus, le reti di distribuzione dei clock e dei segnali di controllo (per esempio, i segnali di reset e di set). Questi segnali controllano il funzionamento di molte porte logiche, quindi il loro fan-out è necessariamente elevato. Altri esempi di fan-out elevati si trovano nelle memorie, nelle quali un numero elevato di celle di memoria è connesso a un numero ristretto di linee dati e di controllo; le capacità di questi nodi sono facilmente dell'ordine di diversi picofarad. Il caso peggiore si verifica però quando i segnali devono essere trasmessi al di fuori del chip. In questo caso, il carico è costituito dalle connessioni del package e della scheda a circuiti stampati e dalla capacità di ingresso dei circuiti integrati o dei componenti connessi. I carichi al di fuori del chip possono raggiungere i 50 pF, che è un valore molte migliaia di volte superiori ai normali carichi all'interno del chip. Pilotare questi nodi con velocità sufficienti diviene uno dei problemi progettuali più critici.

I segreti principali per pilotare efficientemente i carichi capacitivi notevoli sono già stati discussi nel Capitolo 5 e si possono riassumere in due concetti fondamentali:

- Un opportuno **dimensionamento dei transistor** è fondamentale quando si tratta con carichi elevati.
- **Suddividere lo stadio di uscita** in catene di buffer di dimensioni che aumentano progressivamente aiuta a gestire fan-out elevati.

Materiale protetto da copyright

Per chiarezza vale la pena ripetere alcune conclusioni importanti dedotte da quell'analisi:

- Quando si ottimizza il sistema rispetto alle prestazioni, il ritardo di un buffer a più stadi dovrebbe essere **suddiviso equamente tra i vari stadi**.
- Se è possibile scegliere liberamente il numero di stadi, un **fattore di fan-out di circa 4 per ogni stadio** porta al ritardo minimo nei processi a semiconduttore contemporanei. Scegliere fattori maggiori entro certi limiti non degrada eccessivamente le prestazioni, portando al contempo a benefici significativi in termini di area.

Nei paragrafi seguenti approfondiremo ulteriormente questa tematica: ci concentreremo particolarmente sul pilotaggio di capacità molto elevate, quali quelle che si presentano nel caso si esca dal chip; presenteremo anche alcuni circuiti per pilotare il carico capacitivo (*driver*) speciali e piuttosto utili.

Pilotare capacità al di fuori del chip Come affermato nel Capitolo 2, l'aumento della complessità negli odierni circuiti integrati si traduce in un'esplosione del numero di pin di ingresso/uscita: infatti i package con più di 1 000 pin sono ormai una realtà. Questo pone specifiche stringenti sui pad di bonding in termini di immunità al rumore. La commutazione simultanea di molti pad, ciascuno dei quali pilota un condensatore notevole, dà origine a elevate correnti di transitorio e a variazioni di tensione sulle linee di alimentazione e massa. Questo riduce i margini al rumore e degrada l'integrità del segnale, come sarà evidente più avanti in questo capitolo.

Allo stesso tempo, la riduzione delle geometrie della tecnologia fa diminuire le capacità da pilotare all'interno del chip, mentre quelle all'esterno rimangono all'incirca costanti, tipicamente da 10 a 20 pF. Di conseguenza il fattore di fan-out complessivo F del buffer di un pin di uscita subisce un aumento netto con la riduzione delle geometrie della tecnologia. In un approccio progettuale coerente, ci aspettiamo che i ritardi di propagazione all'esterno del chip scalino nello stesso modo di quelli all'interno del chip. Questo rende ancora più critica la progettazione dei buffer dei pad, le cui problematiche sono ben illustrate dal semplice esempio che segue.

Esempio 9.3 Progettazione di un buffer di uscita

Consideriamo il caso in cui un invertitore su chip a dimensione minima debba pilotare un condensatore esterno al chip C_L=20 pF. C_i è circa uguale a 2.5 fF per una normale porta logica in un processo CMOS da 0.25 μm. Questo corrisponde a un t_{p0} di circa 30 ps. Il fan-out complessivo F (il rapporto tra C_L e C_i) è uguale a 8000 e richiede senza dubbio un buffer a stadi multipli. Dall'Eq. 5-36, ipotizzando $\gamma = 1$, ricaviamo che sette stadi corrispondono a una soluzione quasi ottima, con un fattore f di 3.6 e tempo di propagazione totale di 0.97 ns. Per un rapporto tra PMOS e NMOS di 1.9 (il rapporto ottimo ricavato per il nostro processo standard nell'Esempio 5.6) e una dimen-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

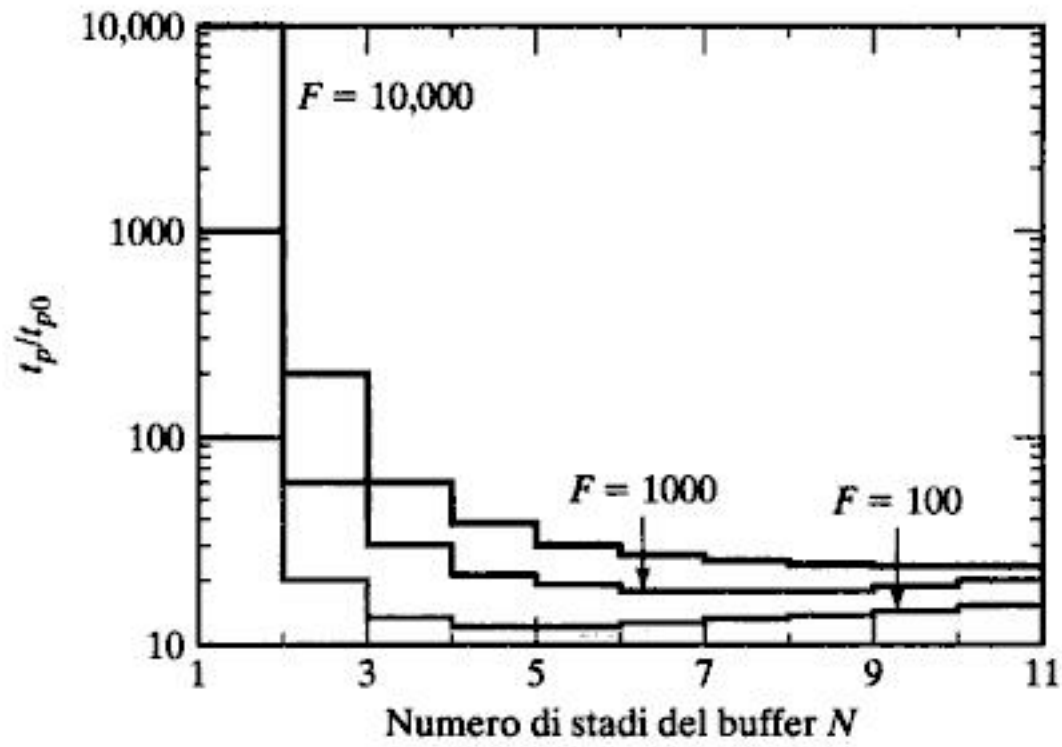


Figura 9-8 t_p/t_{p0} in funzione del numero di stadi del buffer per diversi valori del fan-out globale F.

Usando la larghezza totale dei transistor come figura di merito, possiamo fare una proiezione del risparmio di area conseguente a un maggiore fattore di scaling dei vari stadi del buffer. Assumendo che l'area dell'invertitore minimo sia A_{min} e che moltiplicare la larghezza di un transistor per un coefficiente f implichi un aumento di area dello stesso fattore, possiamo ricavare l'area totale del buffer in funzione di f:

$$\begin{aligned} A_{buffer} &= (1+f+f^2+\ldots+f^{N-1})A_{min} \\ \left(\frac{f^N-1}{f-1}\right)A_{min} &= \frac{F-1}{f-1}A_{min} \end{aligned} \tag{9.5}$$

In sostanza, l'area del buffer è all'incirca inversamente proporzionale al fattore f. Scegliere per f valori più elevati, aiuta a ridurre sostanzialmente il costo in termini di area.

Vale anche la pena di esaminare la questione della dissipazione di energia del buffer esponenziale. Sebbene semplicemente per commutare la capacità di carico sia necessaria una quantità di energia pari a $C_L V_{DD}^2$, il buffer stesso dissipa energia per commutare le sue capacità interne (ignorando l'energia di corto circuito). Questo contributo di energia addizionale può essere approssimato dall'espressione seguente:

$$\begin{aligned} E_{buffer} &= (1+f+f^2+\ldots+f^{N-1})(1+\gamma)C_i V_{DD}^2 \\ &= \left(\frac{F-1}{f-1}\right)(1+\gamma)C_i V_{DD}^2 \approx \frac{2C_L}{f-1}V_{DD}^2 \end{aligned} \tag{9.6}$$

Questo significa che per pilotare velocemente una capacità elevata (cioè usando il fattore di scala ottimo di 3.6) è necessario consumare l'80% in più di energia rispetto al caso senza buffer. Per capacità di carico elevate questo è un aumento notevole. Scostarsi dal valore ottimale del fattore di scala può essere di aiuto per ridurre questa dissipazione addizionale.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

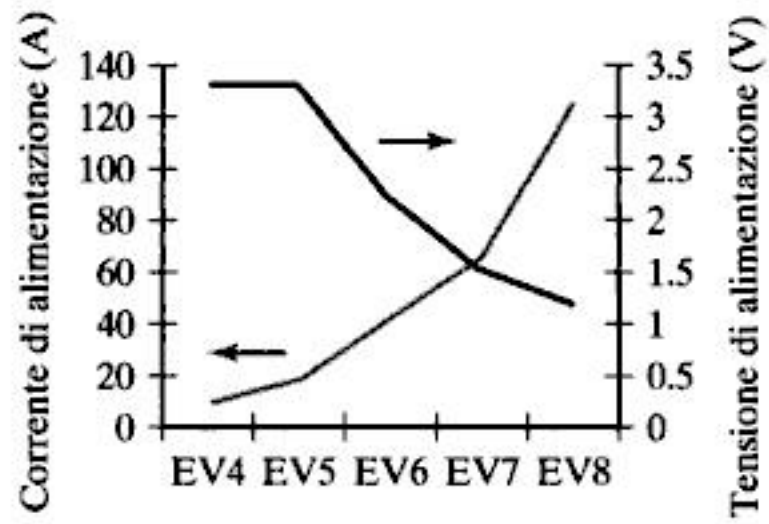


Figura 9-13 Evoluzione della corrente assorbita dall'alimentazione e della tensione di alimentazione per generazioni differenti dei microprocessori ad alte prestazioni della famiglia Alpha della Compaq. Nonostante la riduzione della tensione di alimentazione, per le nuove generazioni è necessario sopportare una corrente totale maggiore di 100 A [Herrick00].

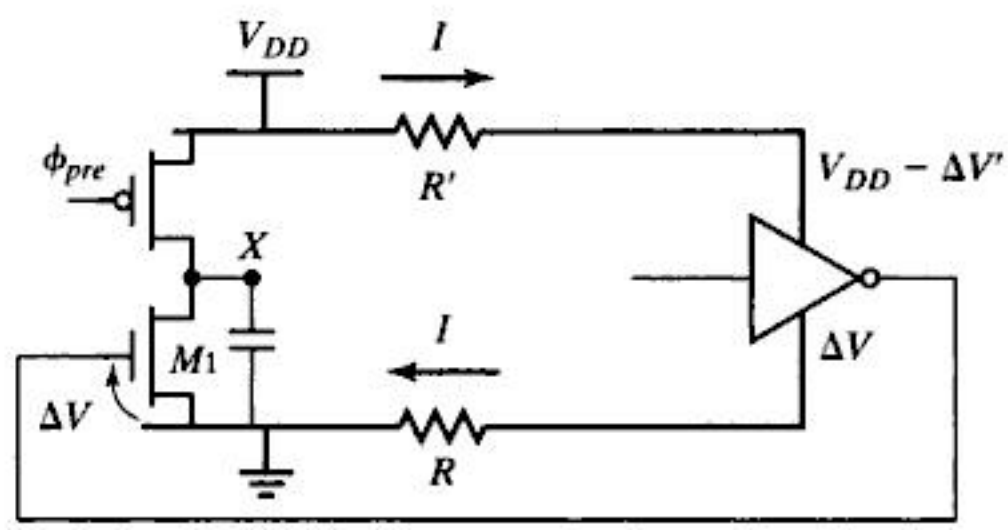


Figura 9-14 La caduta di tensione sulle piste di alimentazione causa la riduzione dei margini di rumore.

che una pista di alluminio può sopportare, a causa dell'*elettromigrazione*, fenomeno discusso nel prossimo paragrafo. Ipotizzando una resistenza di 0.05 Ω/quadro, la resistenza di questa pista (per μm di larghezza) è pari a 1 kΩ. Una corrente di 1 mA/μm comporterebbe quindi una caduta di tensione di 1 V. Il diverso valore della tensione di alimentazione comporta la riduzione dei margini di rumore e una variazione dei livelli logici in funzione della distanza dai terminali di alimentazione. Questo fenomeno è illustrato nella Figura 9-14, dove un invertitore posizionato lontano dai terminali di alimentazione e massa viene connesso a un dispositivo più vicino all'alimentazione. La differenza dei livelli logici, causata dalla caduta di tensione *IR* lungo la linea di alimentazione, può causare l'accensione parziale del transistor M_1. Questo può causare a sua volta la scarica accidentale del nodo dinamico precaricato *X*, oppure può causare consumo statico di potenza se la porta connessa in cascata è statica. In definitiva gli impulsi di corrente dovuti alla logica e alla memoria sul chip e ai terminali di ingresso e uscita, provocano delle cadute di tensione sulla rete di distribuzione dell'alimentazione e sono quindi la sorgente principale del rumore sulle piste di alimentazione sul chip. Oltre a causare questi rischi in termini di robustezza, le cadute *IR* sulle reti di alimentazione inficiano anche le prestazioni del sistema, dato che anche una piccola riduzione della tensione di alimentazione può comportare un aumento significativo dei ritardi.

Materiale protetto da copyright

piani metallici inoltre fungono anche da schermi tra i livelli che distribuiscono i dati, riducendo di conseguenza il cross talk. Inoltre sono di aiuto per ridurre l'induttanza su chip. Ovviamente questo approccio è percorribile solo se è disponibile un numero sufficiente di livelli metallici.

Il dimensionamento di queste reti è tutt'altro che banale. La griglia di piste può essere modellata come una rete di resistenze (piste) e generatori di corrente (logica) che contiene milioni di elementi. Molto spesso esistono numerosi percorsi tra i terminali di alimentazione e le connessioni all'alimentazione di un modulo o di una porta logica del chip. Mentre, in genere, la corrente fluisce lungo il percorso a resistenza minima, il flusso preciso dipende anche da fattori quali l'assorbimento di corrente di moduli vicini che condividono la stessa rete. L'analisi è ulteriormente complicata dal fatto che i picchi di corrente assorbiti dai vari moduli sono distribuiti nel tempo: la caduta *IR* è un fenomeno dinamico. La maggiore sollecitazione della griglia di alimentazione è dovuta principalmente a eventi di commutazione contemporanei, quali quelli causati dai buffer dei clock e dei bus. La richiesta di corrente dalla rete di alimentazione solitamente raggiunge un picco dopo la commutazione di un clock o quando commuta un driver di grosse dimensioni. D'altra parte, un'analisi al caso peggiore che somma tutti i possibili picchi di corrente porta molto spesso a un sovradimensionamento notevole delle piste.

Risulta quindi evidente come gli strumenti per la progettazione assistita al calcolatore siano un compagno indispensabile nel progettare le reti di distribuzione delle alimentazioni. Data la complessità dei circuiti integrati contemporanei, un'analisi a livello di transistor delle specifiche richieste all'intera rete di distribuzione non è fattibile. Allo stesso tempo non si arriva nemmeno a stime accurate suddividendo il problema in vari sottoblocchi, poiché ogni cambiamento in una sezione tende ad avere delle conseguenze globali. La Figura 9-16 riporta la caduta di tensione *IR* simulata per un circuito digitale complesso. Una prima implementazione (vedi Figura 9-16a) risente di una caduta inaccettabile nel modulo in alto a destra, perché solamente la porzione superiore della griglia alimenta il grosso buffer in cima. Le porzioni inferiori del modulo non sono connesse direttamente alla griglia: la semplice aggiunta di una connessione alla rete risolve in gran parte il problema, come mostra la Figura 9-16b.

In definitiva non è possibile ottenere una previsione accurata del rischio *IR* a meno che il chip completo non venga analizzato come una singola entità. Qualsiasi strumento usato a questo scopo deve quindi essere in grado di analizzare griglie con milioni di resistenze. Fortunatamente al momento sono disponibili alcuni strumenti efficaci e accurati per l'analisi delle griglie di alimentazione [Cadence-Power, RailMill]. Questi combinano un'analisi dinamica delle richieste in termini di corrente da parte dei vari moduli circuitali con un modello dettagliato della rete di alimentazione. Questi strumenti sono oramai indispensabili nel flusso di progettazione.

9.3.2 Elettromigrazione

La densità di corrente (corrente per unità di area) in una pista metallica è limitata da un effetto chiamato *elettromigrazione*. Una corrente *diretta* che scorre in una pista metallica per

Materiale protetto da copyright

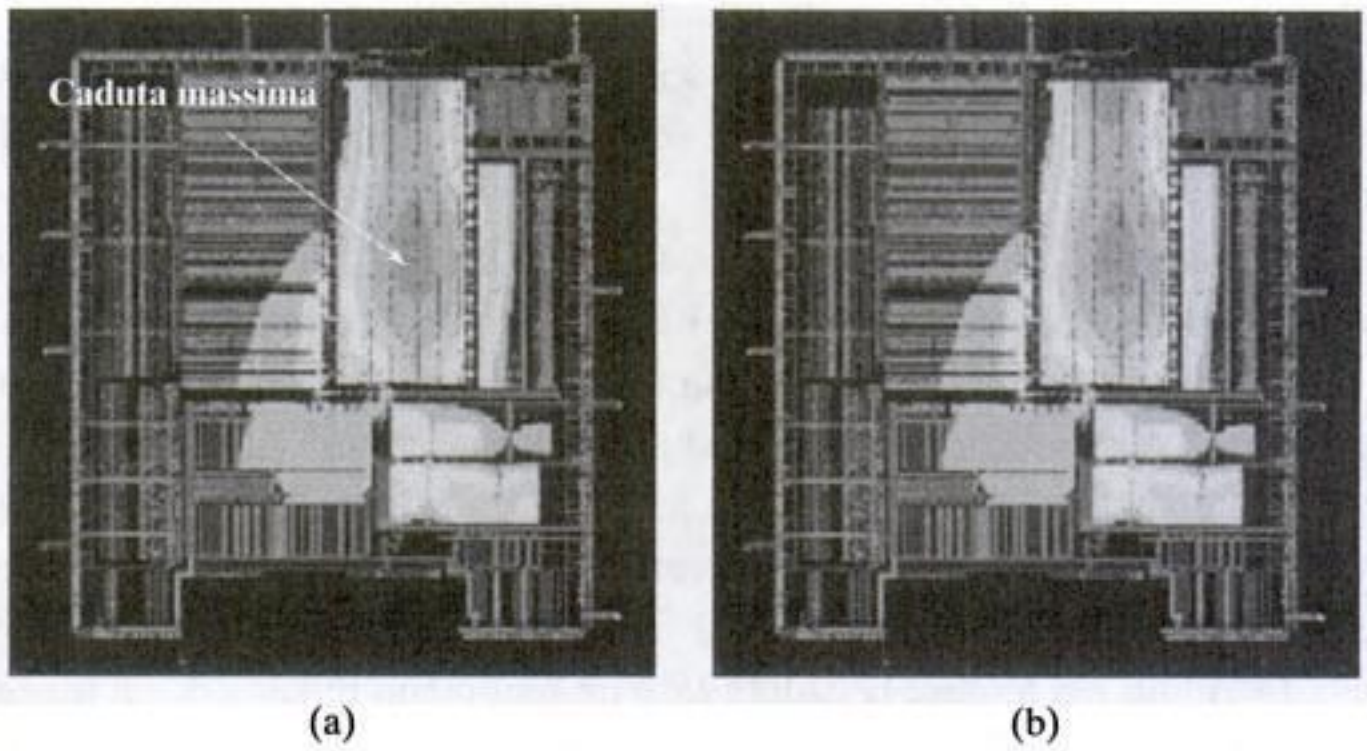
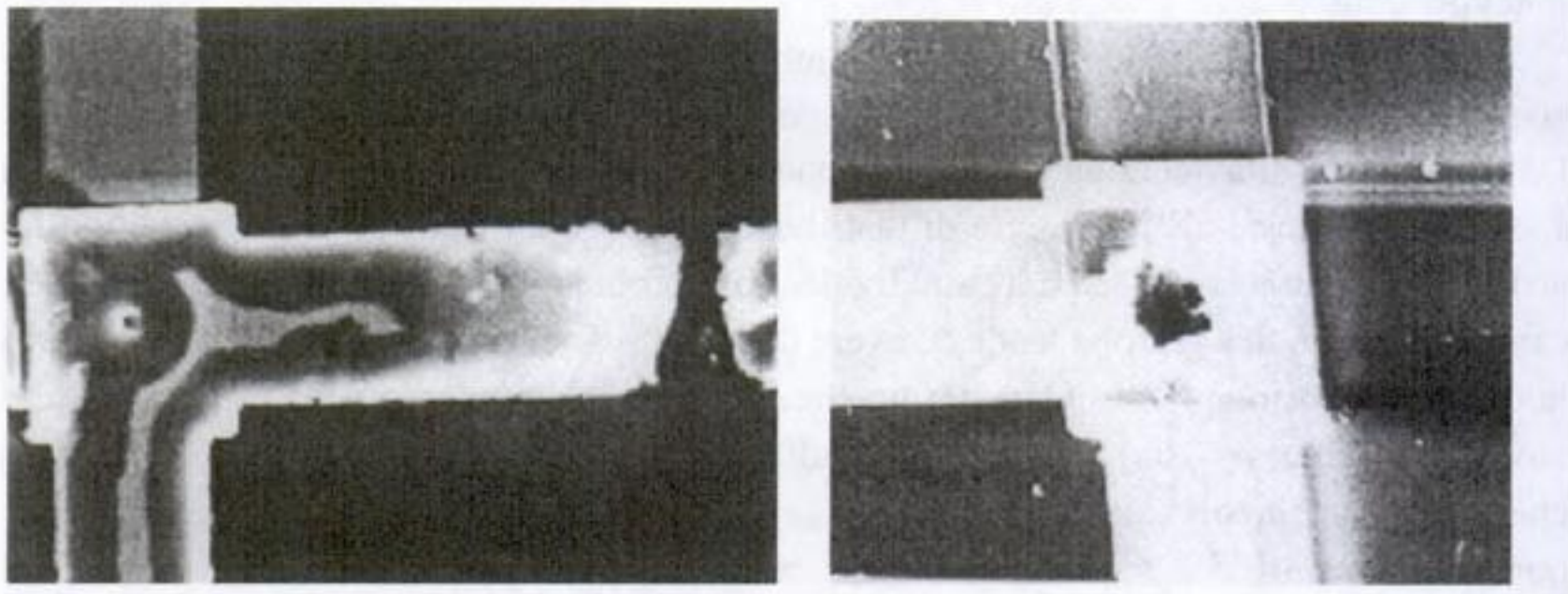


Figura 9-16 Caduta di tensione *IR* simulata in una rete di distribuzione dell'alimentazione per due versioni di un circuito integrato digitale complesso. La scala di grigi è usata per indicare la gravità della caduta di tensione. Aggiungere una connessione nei livelli superiori della rete di alimentazione (b) aiuta a fronteggiare la notevole caduta di tensione nel modulo in alto a destra verificatasi nel sistema iniziale (a) [Cadence-Power]. Vedi anche l'inserto a colori 8.



(a) Interruzione della linea (b) Apertura nell'area di contatto

Figura 9-17 Danni dovuti a elettromigrazione. (Gentile concessione di N. Cheung e A. Tao, U.C. Berkeley.)

un certo periodo di tempo causa il trasporto degli ioni del metallo. Di conseguenza questo causa la rottura della pista o il corto circuito della stessa con un'altra pista. Questo tipo di danno si verifica solo se il dispositivo è stato usato per un certo tempo. Alcuni esempi di danni causati dall'elettromigrazione sono mostrati nella Figura 9-17. Notiamo come la prima foto mostri chiaramente il formarsi di un rigonfiamento nella direzione in cui scorre la corrente.

Il tasso di elettromigrazione dipende dalla temperatura, dalla struttura cristallina e dalla densità media di corrente; quest'ultimo è l'unico fattore che può essere controllato dal progettista del circuito. Mantenere la corrente al di sotto di 0.5-1 mA/μm normalmente previene la migrazione. Questo parametro può essere usato per determinare la larghezza minima delle

Materiale protetto da copyright

piste delle reti di distribuzione delle alimentazioni. Le piste di segnale normalmente trasmettono dei segnali di corrente variabili e sono meno suscettibili di migrazione: il flusso bidirezionale degli elettroni tende a riparare qualsiasi danno causato alla struttura cristallina. Molte industrie impongono ai loro progettisti delle linee guida severe rispetto al dimensionamento delle piste, sulla base di misure sperimentali e dell'esperienza accumulata; risultati di ricerca hanno dimostrato che molte di queste regole tendono a essere eccessivamente conservative.

Regola di progetto

Gli effetti dell'elettromigrazione sono proporzionali al flusso medio di corrente nella pista, mentre le cadute di tensione *IR* dipendono dal picco della corrente. ■

A livello di tecnologia è possibile adottare un certo numero di precauzioni per ridurre il rischio di elettromigrazione. Una possibilità consiste nell'aggiungere elementi leganti (quali Cu o Tu) all'alluminio per prevenire il movimento degli ioni di Al. Un'altra possibilità è quella di controllare la granularità degli ioni. Anche l'introduzione di nuovi materiali per le interconnessioni è di grande aiuto: per esempio, l'aspettativa di vita per una pista aumenta di un fattore 100 quando si usano interconnessioni in rame invece che in alluminio.

9.3.3 Resistenza e prestazioni: ritardo *RC*

Nel Capitolo 4 abbiamo dimostrato che il ritardo di una pista cresce quadraticamente con la sua lunghezza: raddoppiare la lunghezza di una pista porta quindi a un aumento del ritardo di un fattore 4. Il ritardo introdotto da piste lunghe è quindi dominato dall'*effetto RC*. Questo sta divenendo un problema sempre più grave nelle tecnologie moderne, dove la lunghezza media delle connessioni sta aumentando (Figura 4-27) e, allo stesso tempo, il ritardo medio delle singole porte logiche sta diminuendo. Questo porta alla strana situazione in cui possono essere necessari più cicli di clock affinché un segnale si propaghi da un estremo all'altro del chip [Dally01]. In queste circostanza diviene sempre più difficile garantire una sincronizzazione accurata e funzionalità corrette. In questo paragrafo, esaminiamo ora alcune tecniche progettuali che possono essere di aiuto per gestire i ritardi causati dalla resistenza di una pista.

Materiali migliori per le interconnessioni

Una prima possibilità per ridurre i ritardi *RC* consiste nell'usare materiali migliori per le interconnessioni, quando siano disponibili. L'introduzione di siliciuri e rame è stata di aiuto per ridurre la resistenza rispettivamente del polisilicio e delle piste di metallo, mentre adottare materiali dielettrici con minore permettività permette di ridurre la capacità. Sia il rame che i dielettrici a bassa permettività sono di uso comune nelle tecnologie CMOS avanzate (a cominciare dalla generazione delle tecnologie da 0.18 μm). Tuttavia il progettista deve tenere presente che questi nuovi materiali forniscono solamente un beneficio temporaneo per una o due generazioni, ma non risolvono in modo fondamentale il problema del ritardo delle piste lunghe: tecniche progettuali innovative sono spesso l'unico modo per affrontarlo.

Materiale protetto da copyright

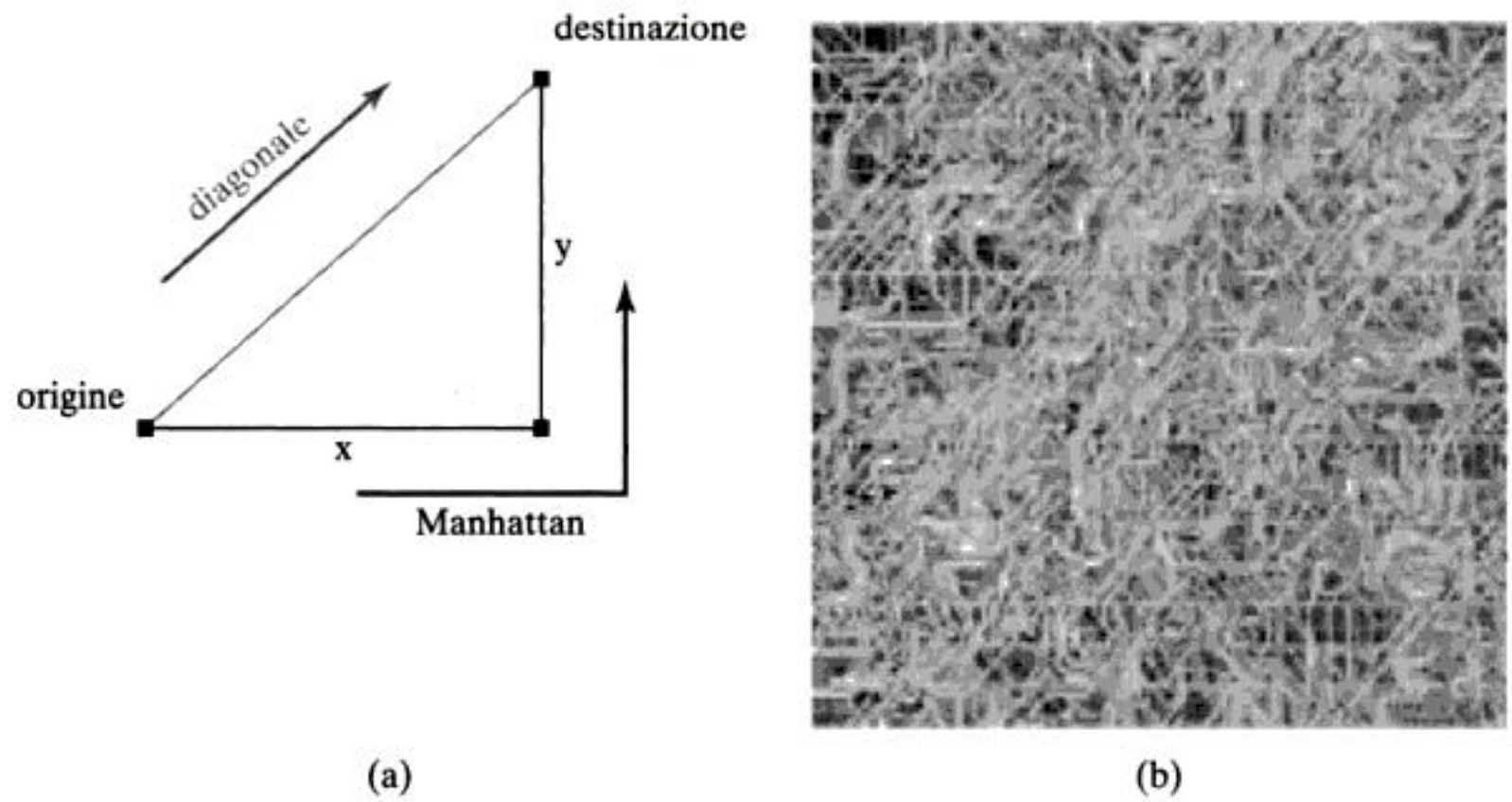


Figura 9-19 Instradamento *Manhattan* a confronto con quello diagonale. (a) Lo stile Manhattan predilige le connessioni lungo gli assi ortogonali, mentre l'instradamento diagonale permette linee a 45°; (b) esempio di un layout che usa le linee a 45°.

lunghezza media delle piste, poiché la congestione dell'instradamento viene ridotta e le interconnessioni possono seguire facilmente un cammino diretto tra la sorgente e la destinazione. Tuttavia, lo stile "Manhattan" che è spesso usato negli odierni strumenti per la stesura delle connessioni, comporta un aumento significativo della lunghezza; aspetto che è spesso trascurato. Secondo lo stile Manhattan, le interconnessioni vengono dapprima stese secondo una tra due direzioni privilegiate e in seguito si esegue la parte di connessione nell'altra direzione (vedi Figura 9-19a). Realizzare la connessione lungo la diagonale permetterebbe una notevole riduzione della lunghezza totale della pista, fino al 29% nel caso migliore. Per ironia del destino, le linee a 45° erano estremamente comuni agli inizi della progettazione dei circuiti integrati, ma divennero presto fuori moda a causa di problemi di complessità, conseguenze per gli strumenti di progettazione e problemi inerenti alla realizzazione delle maschere. Recentemente è stato dimostrato che tutti questi problemi possono essere gestiti in modo adeguato e che le linee a 45° sono assolutamente fattibili [Cadence-X]. Le conseguenze sulle interconnessioni sono piuttosto tangibili: una riduzione del 20% della lunghezza delle piste. Questo a sua volta si traduce in maggiori prestazioni, minore dissipazione di potenza e minore area dei chip. Le potenzialità in termini di densità delle strutture di interconnessione che vanno oltre al limite delle due direzioni dello stile puramente Manhattan sono dimostrate nel campione di layout mostrato nella Figura 9-19b.

Impiego di ripetitori

L'approccio più comune per ridurre il ritardo di propagazione delle piste lunghe è quello di introdurre dei buffer intermedi (noti anche come *ripetitori*) lungo la linea di interconnessione (vedi Figura 9-20). Ridurre la lunghezza di una linea di interconnessione di un fattore m, ne riduce il ritardo di propagazione in modo quadratico e questo è vantaggioso persino conside-

Materiale protetto da copyright

Qui γ rappresenta il rapporto tra le capacità intrinseche di uscita e di ingresso, secondo quanto definito nell'Eq. (5.29). Ponendo $\partial t_p/\partial m$ e $\partial t_p/\partial s$ a 0 otteniamo i valori ottimi per m, s e $t_{p,\,min}$.

$$
\begin{aligned}
m_{opt} &= L\sqrt{\frac{0{,}38rc}{0{,}69R_dC_d(\gamma+1)}} = \sqrt{\frac{t_{ppista(senzabuffer)}}{t_{p1}}} \\
s_{opt} &= \sqrt{\frac{R_dc}{rC_d}} \\
t_{p,\,min} &= (1{,}38 + 1{,}02\sqrt{1+\gamma})L\sqrt{R_dC_drc}
\end{aligned}
\tag{9.10}
$$

Qui $t_{p1} = 0.69R_dC_d(\gamma + 1) = t_{p0}(1 + 1/\gamma)$ rappresenta il ritardo di un invertitore con fan-out 1 ($f = 1$). L'Eq. (9.10) dimostra chiaramente come l'inserimento dei ripetitori renda il ritardo lineare rispetto alla lunghezza. Osserviamo anche che **per una data tecnologia e un dato livello di interconnessione, esiste una lunghezza ottima dei segmenti di pista tra i ripetitori.** Questa *lunghezza critica* è data dall'espressione seguente:

$$
L_{crit} = \frac{L}{m_{opt}} = \sqrt{\frac{t_{p1}}{0{,}38rc}} \tag{9.11}
$$

Si può dimostrare che **il ritardo di un segmento di lunghezza critica** è sempre uguale a

$$
t_{p,\,crit} = \frac{t_{p,\,min}}{m_{opt}} = 2\left(1 + \sqrt{\frac{0{,}69}{0{,}38(1+\gamma)}}\right)t_{p1} \tag{9.12}
$$

ed **è indipendente dal livello di connessione**. Per il valore tipico di $\gamma = 1$, abbiamo $t_{p,\,crit} = 3.9t_{p1} = 7.8t_{p0}$. Inserire dei ripetitori per ridurre il ritardo di una pista ha senso solamente quando la lunghezza di quest'ultima è almeno il doppio della lunghezza critica.

Esempio 9.7 Minimizzazione del ritardo di una pista (rivisitazione)

Nel Capitolo 5 (Esempio 5.5), abbiamo dimostrato che il t_{p1} di un invertitore a dimensione minima nel nostro processo CMOS da 0.25 µm vale 32.5 ps con un valore (medio) di R_d e C_d pari rispettivamente a 7.8 kΩ e 3 fF e $\gamma = 1$. Per una pista in Al-1 (c = 110 aF/µm; $r = 0.075$ Ω/µm), questo implica un fattore di dimensionamento del ripetitore ottimo $s_{opt} = 62$. Inserendo 31 di questi invertitori ingranditi, il ritardo minimo della pista da 10 cm dell'Esempio 9.6 assume il valore più realistico di 3.9 ns. Questo in contrasto al caso dell'inserimento di 31 ripetitori a dimensioni minima, che avrebbe in realtà fatto aumentare il ritardo fino a 61 ns a causa della scarsa corrente erogata dai ripetitori di dimensione minima.

La lunghezza critica per l'Al-1 vale 3.2 mm, mentre è pari a 54 µm e 8.8 mm rispettivamente per polisilicio e Al-5.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

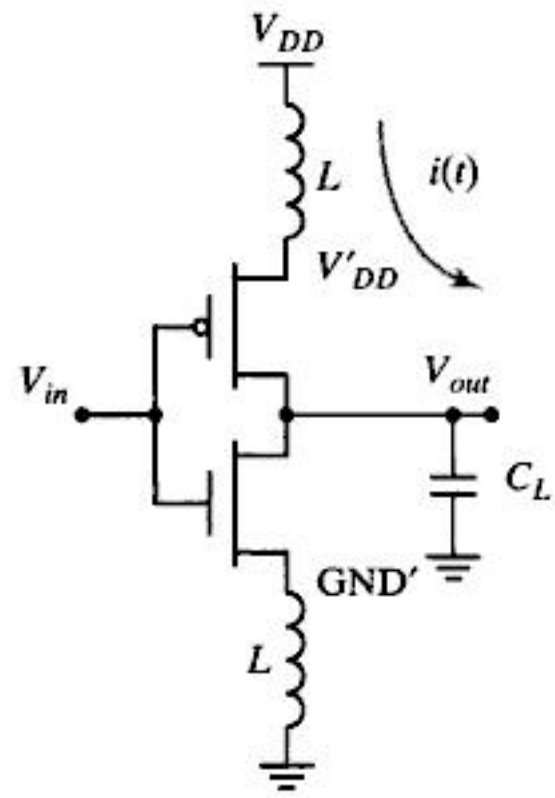


Figura 9-22 Accoppiamento induttivo tra le tensioni di alime ntazione esterne e interne.

verso le connessioni di ingresso/uscita può subire transizioni veloci che causano cadute di tensione così come oscillazioni e sovraelongazioni; fenomeni che non si manifestano nei circuiti *RC*. Inoltre a velocità di commutazione superiori, si manifestano anche gli effetti di propagazione d'onda e linea di trasmissione; entrambi gli effetti sono analizzati in questo paragrafo, per proporre poi delle soluzioni progettuali.

9.4.1 Induttanze e robustezza: cadute di tensione

In corrispondenza di qualsiasi transitorio di commutazione, il circuito assorbe (o inietta) corrente elettrica dalle linee di alimentazione per caricare (o scaricare) le sue capacità, come mostrato nella Figura 9-22. Entrambe le connessioni a V_{DD} e V_{SS} sono portate all'esterno del chip attraverso fili di bonding e piedini e quindi presentano un'induttanza serie tutt'altro che trascurabile. Una variazione di corrente provoca quindi una differenza di tensione tra le alimentazioni esterne e quelle interne (V'_{DD}, GND′). Questo problema è particolarmente significativo nel caso dei pad di uscita, in cui la necessità di pilotare le capacità esterne elevate implica l'assorbimento di correnti significative. La variazione delle tensioni di alimentazione interne degrada i livelli logici e quindi causa una riduzione dei margini al rumore.

Esempio 9.8 Rumore causato dall'induttanza dei fili di bonding e dai piedini del package

Ipotizziamo che il circuito della Figura 9-22 sia l'ultimo stadio del buffer di un pad di uscita che deve pilotare una capacità di carico di 10 pF con un'escursione logica di 2.5 V. L'invertitore è stato dimensionato in modo che i tempi di salita e discesa dal 10% al 90% del segnale di uscita (t_r, t_f) siano 1 ns. Poiché le connessioni all'alimentazione e alla massa esterne avvengono attraverso i piedini di alimentazione, entrambi questi collegamenti hanno un'induttanza serie L. Nel caso di un package con il tradizionale foro

Materiale protetto da copyright

passante, l'induttanza vale tipicamente 2.5 nH. Per semplificare l'analisi, ipotizziamo dapprima che l'invertitore agisca come un generatore di corrente con un valore costante della corrente di (s)carica della capacità di carico. Per ottenere i tempi di salita e discesa di 1 ns in uscita, è richiesta una corrente media di 20 mA:

$$I_m = (10\ \text{pF} \times (0.9 - 0.1) \times 2.5\ \text{V}) / 1\ \text{ns} = 20\ \text{mA}$$

Questa situazione si verifica nel caso in cui il buffer sia pilotato al suo ingresso da un segnale a gradino molto ripido. La parte sinistra della Figura 9-23 riporta l'andamento temporale simulato della tensione di uscita, della corrente e della tensione nell'induttore per $t_f = 50$ ps. Il brusco cambiamento della corrente provoca nell'induttore un breve impulso di tensione di ampiezza fino a 0.95 V. In effetti la variazione di tensione sarebbe stata maggiore se la caduta stessa non avesse rallentato il transitorio e di conseguenza diminuito la corrente necessaria. Ciò nonostante, una tensione di alimentazione con una tale variazione è inaccettabile.

In realtà la corrente di carica è raramente costante. Una migliore approssimazione consiste nel considerare che la corrente cresca linearmente fino a un valore massimo e quindi decresca fino a zero, ancora con andamento lineare. La distribuzione nel tempo della corrente può in questo caso essere approssimata con un triangolo. Questa situazione si verifica quando il segnale in ingresso al buffer è caratterizzato da tempi di salita e discesa lenti. Usando questo modello, la caduta di tensione (stimata) sull'induttore è data da

$$v_L = L\, di_L/dt = (2.5\ \text{nH} \times 40\ \text{mA})/((1.25/2) \times 1\ \text{ns}) = 133\ \text{mV}.$$

Il picco di corrente (circa) 40 mA discende dal fatto che la quantità totale di carica fornita, ovvero l'area dell'integrale di I_L, è fissa, quindi il picco della distribuzione triangolare di corrente è il doppio rispetto al caso di quella rettangolare. Il fattore a denominatore stima il tempo necessario a commutare da zero al picco di corrente. I risultati di simulazione che si ottengono in questo caso sono riportati nella parte destra della Figura 9-23 per $t_f = 800$ ps. Si può osservare che la distribuzione di corrente diviene effettivamente triangolare con la pendenza limitata dell'ingresso. La caduta di tensione simulata ai capi dell'induttore è minore di 100 mV. Questo suggerisce di evitare segnali che commutano rapidamente in ingresso ai buffer i quali pilotano grosse capacità per ridurre in modo efficace le cadute di tensione sulle induttanze parassite delle linee.

In un circuito reale, un singolo terminale di alimentazione è usato da un numero elevato di porte logiche o buffer di uscita, la cui commutazione contemporanea provoca transitori di corrente e cadute di tensioni ancora peggiori. Di conseguenza la tensione di alimentazione interna differisce in modo significativo da quella esterna. Per esempio, la commutazione contemporanea di 16 buffer di un bus di uscita provocherebbe una caduta di tensione di almeno 1.1 V se i nodi di alimentazione dei buffer fossero tutti connessi allo stesso terminale.

Materiale protetto da copyright

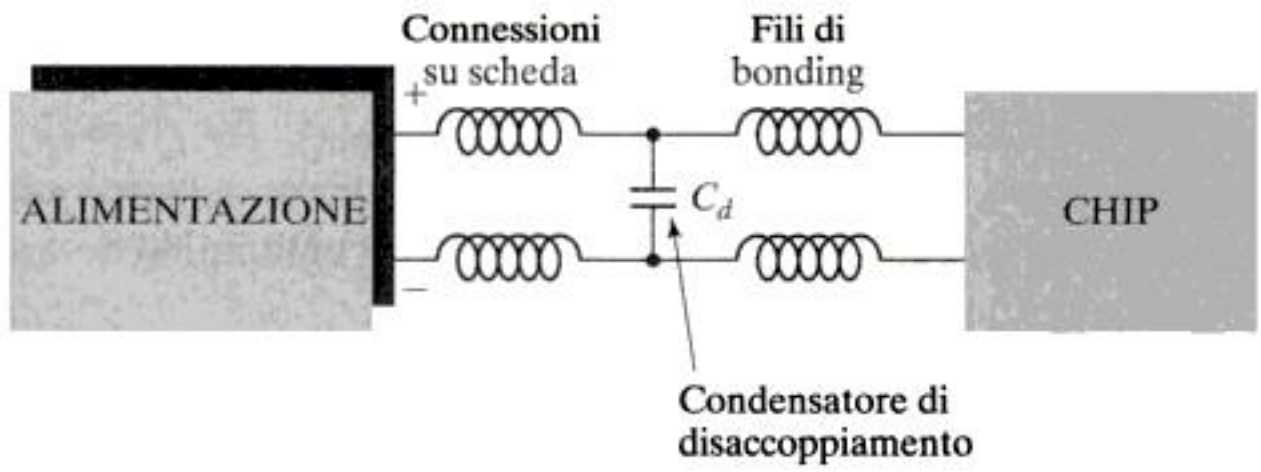


Figura 9-25 I condensatori di disaccoppiamento isolano l'induttanza della scheda da quella del bonding, della pista e del piedino.

chip e usando pasta adesiva si riduce a 0.1 nH, che è da 50 a 100 volte minore di quella dei package standard.

7. **Aggiungere capacità di disaccoppiamento sulla scheda.** Queste capacità, che devono essere aggiunte per ogni terminale di alimentazione, agiscono come alimentazioni locali e aiutano a stabilizzare la tensione di alimentazione vista dal chip. Esse separano l'induttanza del filo di bonding e della pista da quella delle interconnessioni della scheda (vedi Figura 9-25). Il condensatore di disaccoppiamento, combinato con l'induttanza, in realtà si comporta come una rete passa-basso che sopprime le componenti in alta frequenza dei picchi di tensione sulle linee di alimentazione.

Esempio 9.9 Effetto delle capacità di disaccoppiamento

Una capacità di disaccoppiamento di 200 pF è stata aggiunta tra le connessioni di alimentazione del buffer esaminato nell'Esempio 9.8: il suo effetto è mostrato nelle simulazioni della Figura 9-23. Per un segnale di ingresso con tempi di salita da 50 ps, il picco sulle linee di alimentazione è ridotto da 0.95 V (senza capacità di disaccoppiamento) a circa 70 mV. Le forme d'onda dimostrano anche come la corrente necessaria a caricare la capacità di uscita sia inizialmente prelevata dal condensatore. Successivamente la corrente viene gradualmente assorbita dalla rete di alimentazione attraverso gli induttori del bonding e della pista.

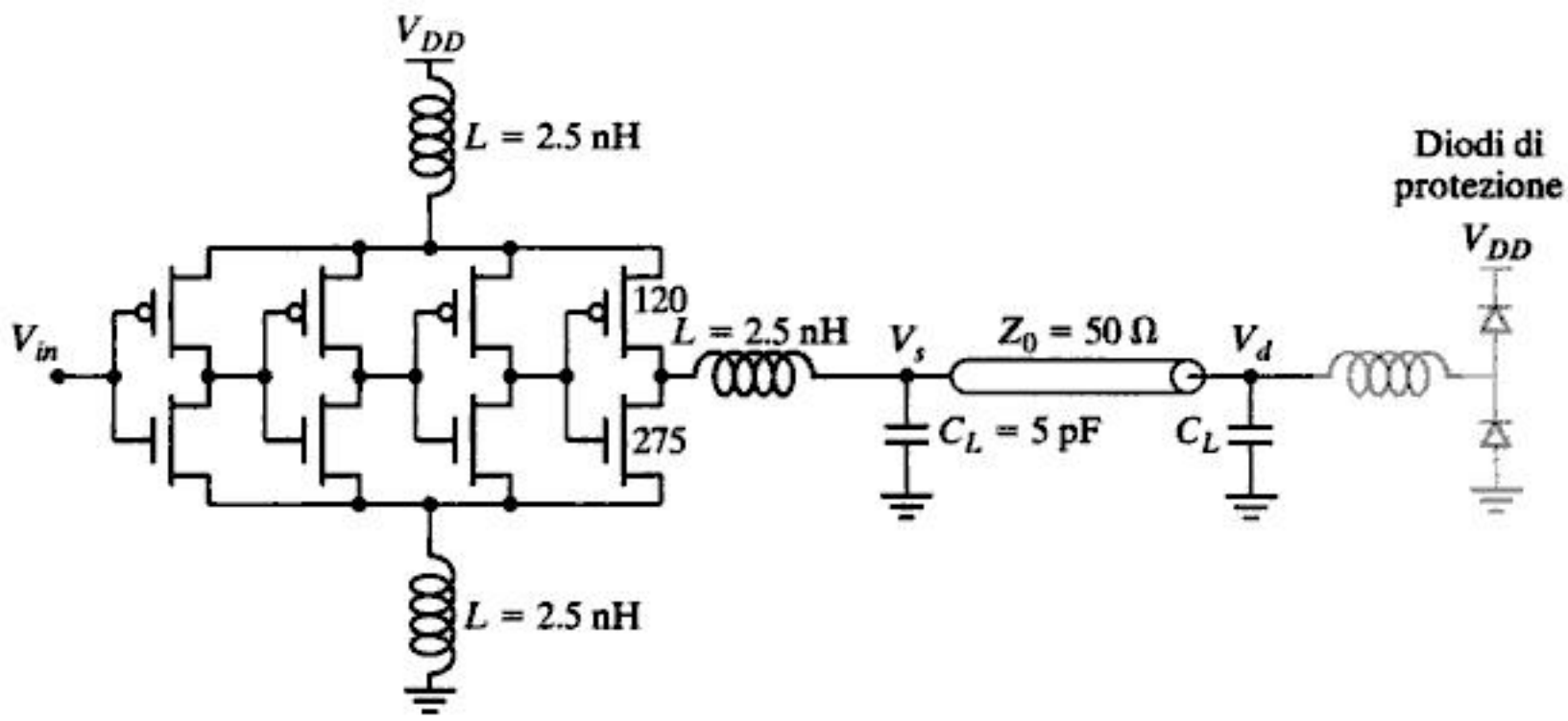
---

8. **Aggiungere capacità di disaccoppiamento sul chip.** Nei circuiti ad alte prestazioni, con velocità di commutazione elevate e transizioni dei segnali ripidi, sta divenendo pratica comune integrare sul chip delle capacità di disaccoppiamento, garantendo tensioni di alimentazione meno rumorose. I condensatori di disaccoppiamento sul chip riducono il picco di assorbimento di corrente al suo valor medio. Per limitare l'oscillazione di tensione a 0.25 V, è necessario predisporre una capacità di circa 12.5 nF per ogni modulo da 50 000 porte logiche nel processo CMOS da 0.25 μm [Dally98]. Queste capacità sono solitamente implementate usando l'ossido sottile di gate; un condensatore a ossido sottile è praticamente un transistor MOS con drain e source cortocircuitati. È ugualmente possibile usare i condensatori di diffusione per tal scopo.

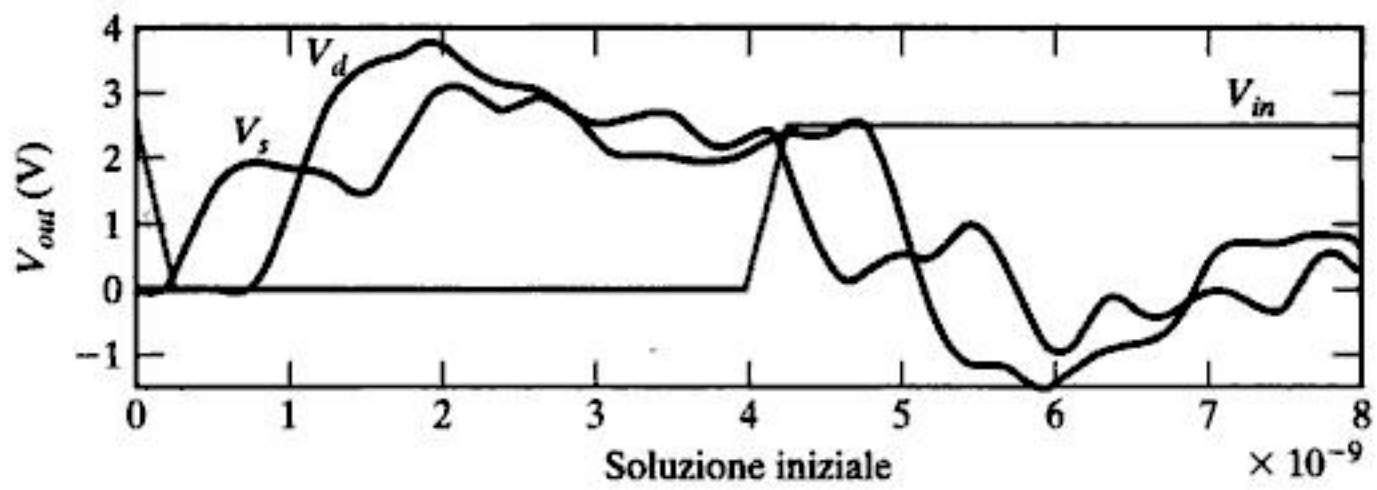
Materiale protetto da copyright

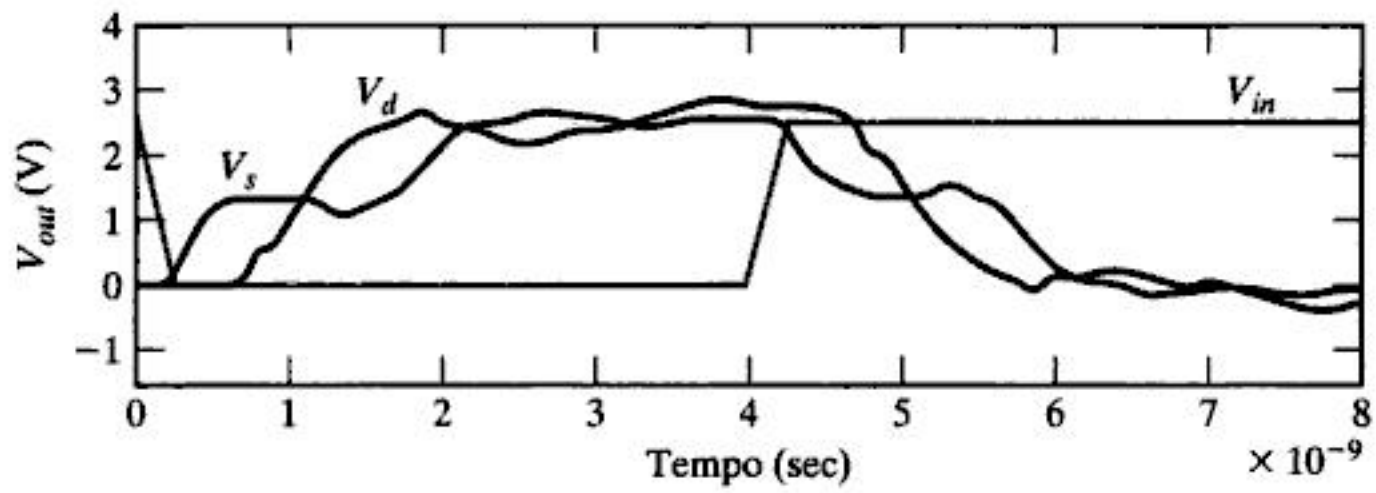
You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Schema circuitale del driver di uscita





Soluzione modificata con adattamento dell'impedenza del driver

(b) Simulazioni al transitorio

Figura 9-30 Simulazione di un driver di uscita per diverse terminazioni.

segnale aiuta a ridurre il ritardo e la dissipazione di potenza quando si devono pilotare piste di elevata capacità. Presenteremo, inoltre, alcune tecniche per limitare l'escursione.

9.5.1 Circuiti a escursione logica ridotta

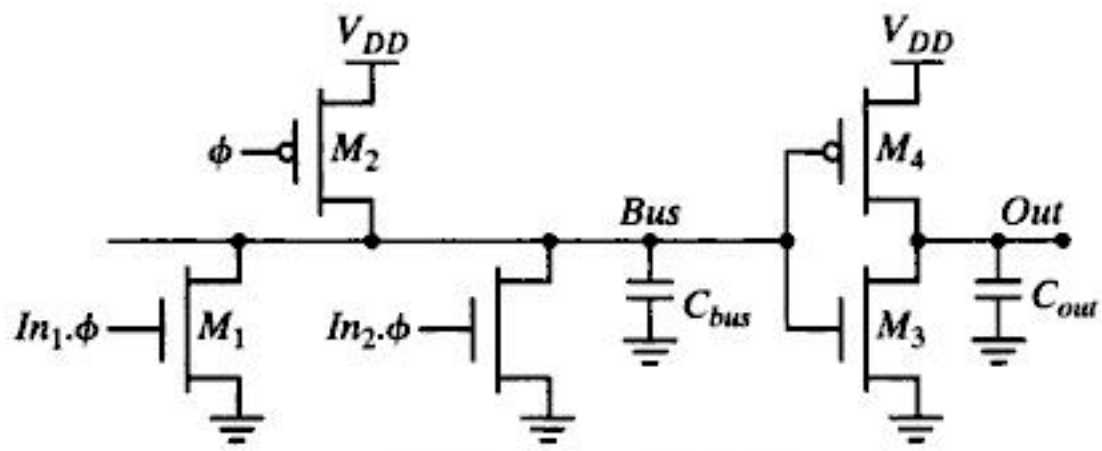
Quello di aumentare le dimensioni del transistor di uscita (e quindi aumentare la corrente media I_{av} durante la commutazione) è solo uno dei modi possibili per compensare il ritardo causato da una capacità di carico elevata. Un approccio alternativo consiste nella riduzione dell'escursione

Materiale protetto da copyright

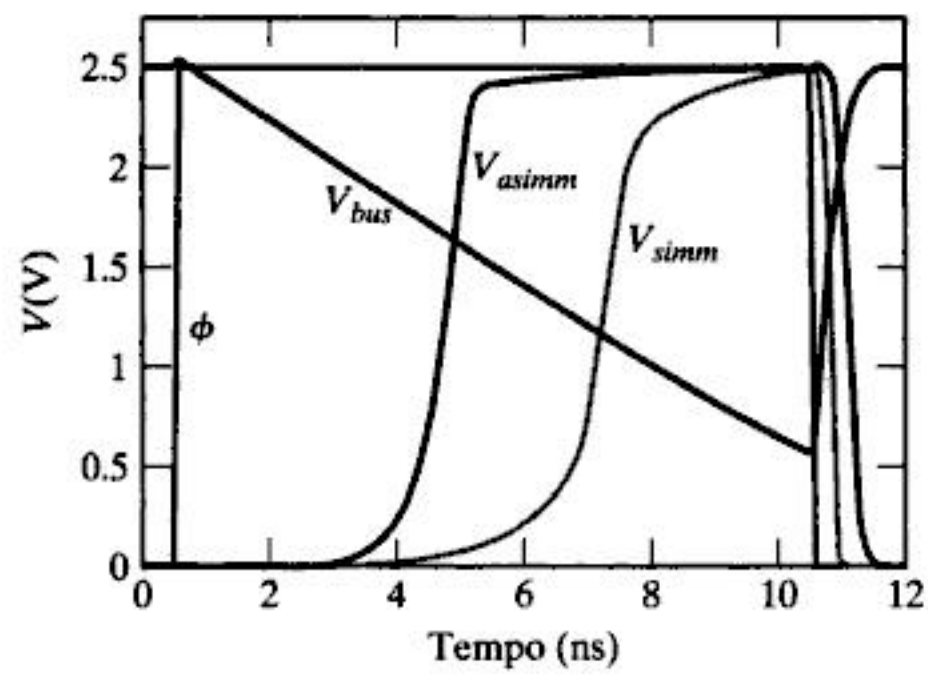
You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



(a) Diagramma circuitale



(b) Risposta simulata degli invertitori di lettura simmetrici e assimetrici. M_4 Ł 15 volte piø largo nel caso asimmetrico. La capacit□ di carico vale 1 pF.

Figura 9-35 Bus a precarica.

Circuiti dinamici a escursione ridotta

Un altro approccio per accelerare la risposta di circuiti con elevato fan-in, quali i bus, consiste nell'usare la precarica, come nell'esempio della Figura 9-35. Quando $\phi = 0$, la pista del bus viene precaricata a V_{DD} attraverso il transistor M_2. Dato che questo dispositivo è condiviso da tutte le porte di ingresso, può essere sufficientemente largo da assicurare una fase di precarica veloce. Quando $\phi = 1$, la capacità del bus viene eventualmente scaricata da uno dei transistor di pull-down. Questa operazione è lenta, perché la grossa capacità C_{bus} deve essere scaricata attraverso un piccolo dispositivo di pull-down, M_1.

Si può ottenere un'accelerazione, a spese dei margini di rumore, partendo dalla considerazione che durante la fase di valutazione tutte le transizioni sul bus sono da alto a basso. Una risposta più veloce si può ottenere spostando verso l'alto la soglia di commutazione dell'invertitore in cascata al bus, il che implica una porta assimetrica. In un invertitore classico, M_3 e M_4 sono dimensionati in modo che t_{pHL} e t_{pLH} siano identici e la soglia di commutazione (V_M) dell'invertitore sia posizionata nell'intorno di $V_{DD}/2$. Questo implica che la tensione del bus V_{bus} deve scendere al di sotto di $V_{DD}/2$ prima che l'uscita dell'invertitore

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

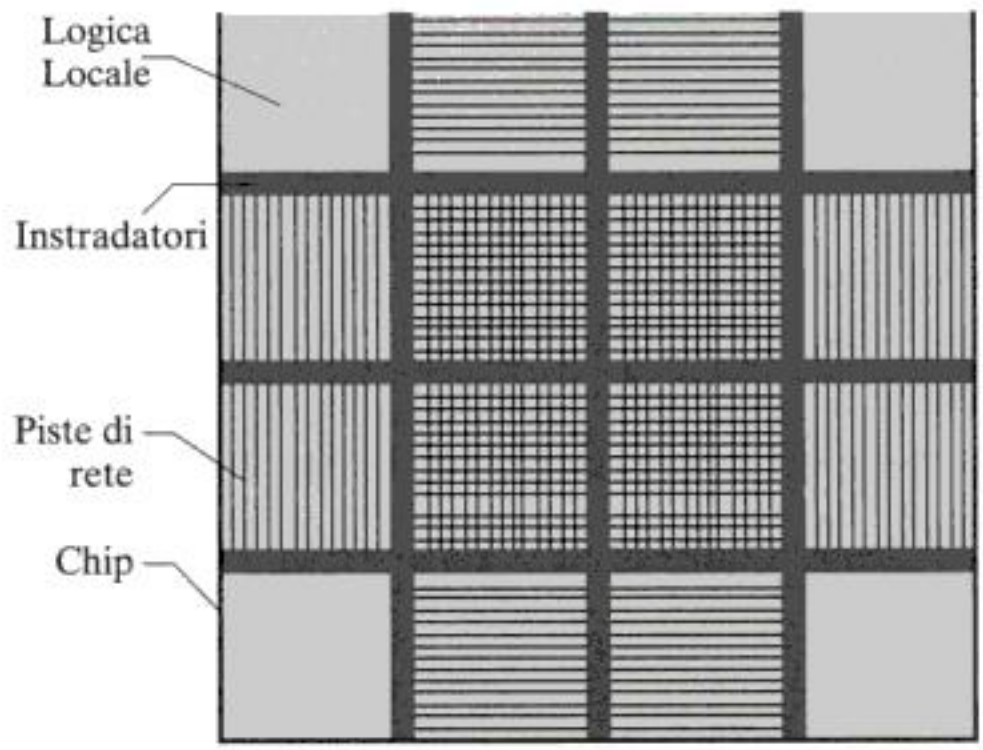


Figura 9-38 L'approccio di rete su chip (network-on-a-chip) combina più processori e le relative reti di interconnessioni sullo stesso chip.

tere dati con un certo livello di "qualità del servizio", ponendo delle specifiche in termini di tasso di uscita dei dati, latenza e correttezza [Sgroi01].

Per fare un esempio, nelle attuali interconnessioni su chip si fa in modo di ottenere margini al rumore tali da garantire che un bit venga sempre trasmesso correttamente. Si può affermare con una certa sicurezza che i sistemi del futuro useranno una tattica più estrema: potrebbero abbandonare completamente l'integrità per ottenere benefici in termini di energia e prestazioni, e quindi ammettere errori sui segnali trasmessi. Questi errori possono poi essere corretti da altri circuiti, progettati per fornire funzioni di correzione di errore e/o ritrasmissione di dati errati.

A un livello di astrazione più elevato, siamo già testimoni della comparsa di "reti" complete su chip, come si vede nella Figura 9-38. Piuttosto che essere instradati da una sorgente verso una destinazione, i dati sono iniettati come pacchetti in una rete completa di cavi, interruttori e instradatori ed è la rete stessa a decidere dinamicamente come e dove instradare questi pacchetti attraverso i suoi segmenti [Dally01]. In definitiva questo è l'unico approccio che può funzionare quando la discrepanza tra le dimensioni dei dispositivi e le distanze su chip diviene macroscopica.

9.7 Riepilogo

In questo capitolo sono state introdotte diverse tecniche per affrontare le conseguenze delle interconnessioni in termini di prestazioni e robustezza nei circuiti integrati digitali. I parassiti introdotti dalle interconnessioni hanno un duplice effetto sul funzionamento del circuito: (1) introducono rumore e (2) aumentano i tempi di propagazione e il consumo di potenza. In sintesi:

- Il *cross talk capacitivo* in reti di piste molto dense riduce la robustezza del sistema e quindi le sue prestazioni. Una progettazione molto attenta che usa strutture regolari ben caratterizzate o strumenti di progettazione automatica avanzati sono irrinunciabili. È

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

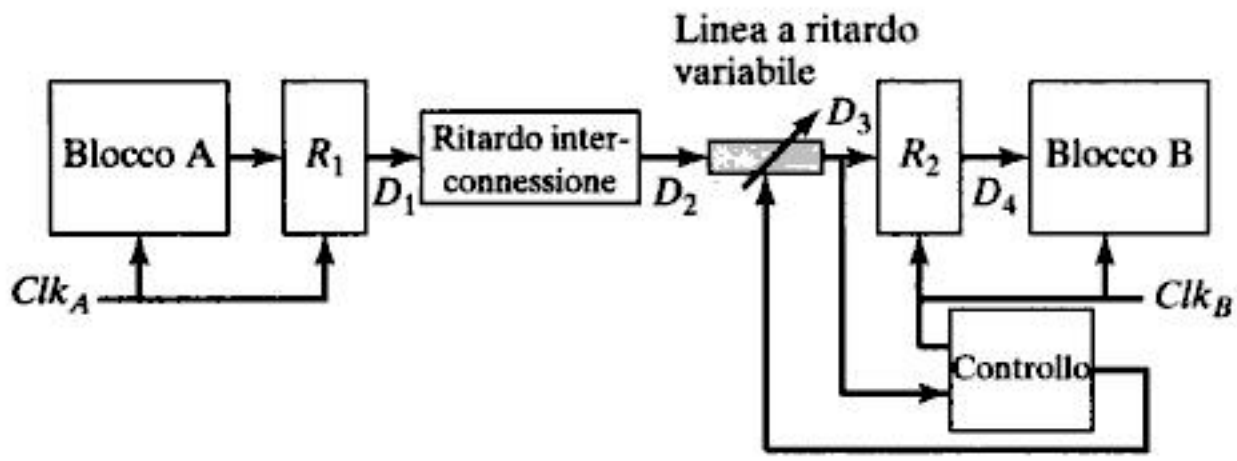


Figura 10-2 Approccio di comunicazione mesocrono che usa una linea a ritardo variabile.

10.2.3 Interconnessione plesiocrona

Un segnale *plesiocrono* è tale quando ha una frequenza che nominalmente è la stessa del clock locale, ma in realtà è leggermente diversa (in greco *plesio* significa "vicino"). Questo fatto comporta che la differenza di fase deriva nel tempo. Questa situazione si può verificare facilmente quando due moduli che interagiscono usano dei clock generati da oscillatori al quarzo indipendenti. Poiché il segnale trasmesso può arrivare al modulo ricevitore a una frequenza diversa rispetto al clock locale, si rende necessario l'uso di uno schema con memoria tampone per garantire la ricezione di tutti i dati. Tipicamente le interconnessioni plesiocrone si incontrano solo in sistemi distribuiti che comprendono canali di comunicazione a lunga distanza, dato che i circuiti sullo stesso chip, o addirittura su una stessa scheda, solitamente sfruttano un unico oscillatore comune per generare i vari clock locali.

Una possibile configurazione per una interconnessione plesiocrona è mostrata nella Figura 10-3. In tale sistema digitale, il modulo sorgente fornisce dati a una certa frequenza incognita C_1, che è plesiocrona rispetto a C_2. L'unità di ripristino della temporizzazione deve ricavare il clock C_3 dalla sequenza di dati e memorizzare gli stessi in un vettore FIFO (*First-in-First-Out*). Di conseguenza, C_3 risulterà sincrono con i dati all'ingresso del vettore e sarà mesocrono rispetto a C_1. Poiché le frequenze dei clock dei moduli sorgente e ricevitore sono diverse, può essere necessario eliminare una parte dei dati se la frequenza di trasmissione è maggiore; al contrario sarà necessario duplicarli se la frequenza di trasmissione è minore di quella del ricevitore. Comunque, assicurandosi che il vettore sia sufficientemente capiente e prevedendo di inizializzare il sistema ogniqualvolta si verifichi una condizione di overflow, è possibile ottenere una comunicazione robusta.

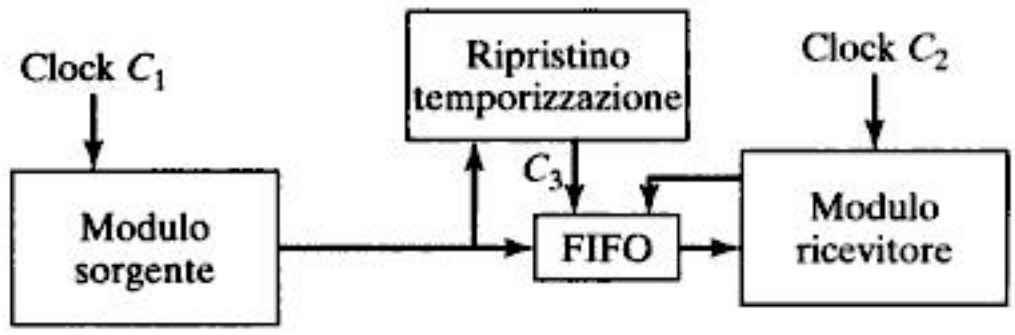


Figura 10-3 Comunicazione plesiocrona che usa una FIFO.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

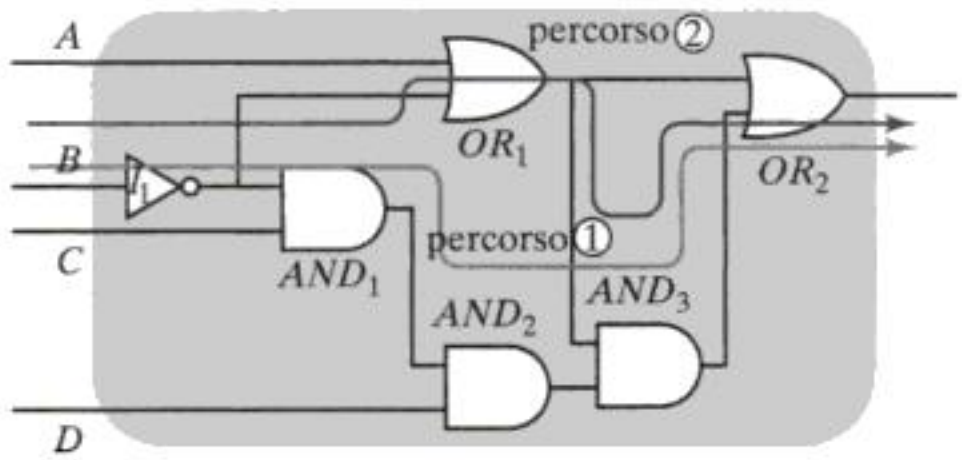


Figura 10-10 Rete logica per la valutazione delle prestazioni.

Il tempo di contaminazione si ottiene facilmente: vale $2t_{gate}$ ed è il ritardo attraverso OR_1 e OR_2. D'altra parte la valutazione del tempo di propagazione al caso peggiore non è altrettanto semplice. A prima vista, può sembrare che il percorso corrispondente al caso peggiore sia il numero ①, il cui ritardo è $5t_{gate}$. Se però analizziamo la dipendenza tra i vari dati, è evidente che il percorso ① non sarà mai attraversato da un flusso di dati ed è quindi detto *percorso falso*. Se $A = 1$, il percorso critico attraversa OR_1 e OR_2. Se $A = 0$ e $B = 0$, il percorso critico attraversa I_1, OR_1 e OR_2 (a cui corrisponde un ritardo pari a $3t_{gate}$). Nel caso in cui $A = 0$ e $B = 1$, il percorso più lungo comprende I_1, OR_1, AND_3 e OR_2. In altre parole, per questa semplice (ma artificiosa) rete l'uscita non dipende dagli ingressi C e D (cioè vi è **ridondanza**). In definitiva quindi l'effettivo tempo di propagazione vale $4t_{gate}$. Una volta ottenuti i tempi di propagazione e di contaminazione, si può calcolare facilmente lo skew massimo e minimo tollerabile.

AVVERTENZA: la valutazione del tempo di propagazione al caso peggiore per reti combinatorie non si può ottenere semplicemente sommando il tempo di propagazione delle singole porte logiche, a causa dei *percorsi falsi*. Il cammino critico dipende fortemente dalla topologia circuitale e della relazione tra i dati.

Jitter

Il fenomeno del *jitter* fa riferimento a variazioni temporali del periodo del clock in un certo punto del chip, cioè il periodo di clock può aumentare o diminuire di ciclo in ciclo: si tratta quindi di un'incertezza temporale in senso stretto, che viene spesso specificata per un dato punto preciso all'interno del chip. Il **jitter** può essere misurato e caratterizzato in molti modi diversi ed **è una variabile aleatoria a media nulla**. Il *jitter assoluto* (t_{jitter}) fa riferimento alla variazione al caso peggiore (valore assoluto) di un fronte di clock in una data posizione, rispetto a un fronte di clock ideale di riferimento. Il *jitter da ciclo a ciclo* (T_{jitter}) fa invece riferimento a deviazioni tempo varianti di un singolo periodo di clock rispetto a un clock di riferimento ideale. Per una data posizione i, è dato da $T^i_{jitter}(n) = t^i_{clk,n+1} - t^i_{clk,n} - T_{CLK}$, dove $t^i_{clk,n+1}$ e $t^i_{clk,n}$ rappresentano rispettivamente il tempo di arrivo al nodo i dei fronti $(n + 1)$-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Variazioni del processo di fabbricazione (2)

I buffer sono componenti fondamentali delle reti di distribuzione dei clock: sono infatti necessari per pilotare il carico offerto sia dai registri sia dalle interconnessioni locali e globali. L'esatta corrispondenza fra i transistor nei buffer posti lungo i vari percorsi di clock è un fattore critico per minimizzare le incertezze temporali. Sfortunatamente, però, a causa delle variazioni del processo di fabbricazione, i parametri dei transistor nei buffer sono diversi lungo percorsi differenti, dando origine a *skew statico*. Vi sono molti parametri che variando contribuisco a questo errore, quali le variazioni dell'ossido (che influiscono su guadagno e soglia), variazioni del livello di drogaggio e variazioni delle dimensioni longitudinali (larghezza e lunghezza). Le variazioni di drogaggio possono influire sulla profondità e sul profilo della giunzione e quindi causare la variazione di parametri elettrici (quali soglia dei dispositivi e capacità parassite).

Anche l'orientazione del polisilicio può influire sui parametri dei dispositivi e quindi è importante mantenere la stessa orientazione per i buffer dei clock lungo tutto il chip. La variazione delle dimensioni del polisilicio è critica perché si traduce direttamente in variazioni della lunghezza di canale dei MOS, alterandone la corrente con cui si pilota il carico e la caratteristica di commutazione. Le variazioni dimensionali si verificano di solito sia a livello di wafer (o intra wafer) sia a livello di chip (o intra chip). Almeno una parte di queste è sistematica e quindi può essere prevista e compensata; la parte casuale però definisce il limite ultimo della corrispondenza tra elementi nominalmente uguali e quindi limita il minimo skew che si può ottenere.

Variazioni delle interconnessioni (3)

A causa delle variazioni delle dimensioni verticali e longitudinali, la capacità e la resistenza delle interconnessioni variano lungo il chip; essendo queste variazioni statiche, danno origine a **skew** tra i diversi percorsi. Una sorgente significativa di variazione dei parametri di un'interconnessione è la variazione dello spessore del *dielettrico tra livelli* (*ILD, Inter-Layer Dielectric*). Nella deposizione delle interconnessioni di alluminio, vengono frapposti degli strati di ossido di silicio tra i vari livelli di metallizzazione. L'ossido è deposto al di sopra di uno strato metallico con cui è stata realizzata una determinata struttura di piste, di cui tipicamente rimane traccia anche a livello dell'ossido, sotto forma di variazioni dello spessore e disuniformità della superficie. Di solito per "planarizzare" la superficie e per rimuovere le tracce della topografia derivanti dai processi di deposizione e incisione (come descritto nel Capitolo 2 e mostrato nella Figura 10-15a) si usa la *lappatura chimica-meccanica* (CMP, *Chemical–Mechanical Polishing*). Sebbene la CMP permetta di ottenere livelli di planarità eccellenti a livello di una singola pista metallica, ha delle limitazioni a livello globale. Tale limitazione è dovuta principalmente al fatto che il grado di lappatura varia a seconda della densità del layout e risente delle diverse strutture geometriche disegnate. La Figura 10-15b mette in evidenza questo effetto: il grado di lappatura è maggiore per le regioni con minore densità spaziale, con conseguente diminuzione dello spessore dell'ossido e quindi aumento della capacità.

La stima e il controllo delle variazioni sono quindi di importanza fondamentale nello sviluppo e nella realizzazione del processo di fabbricazione di semiconduttori. Si sono anche

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

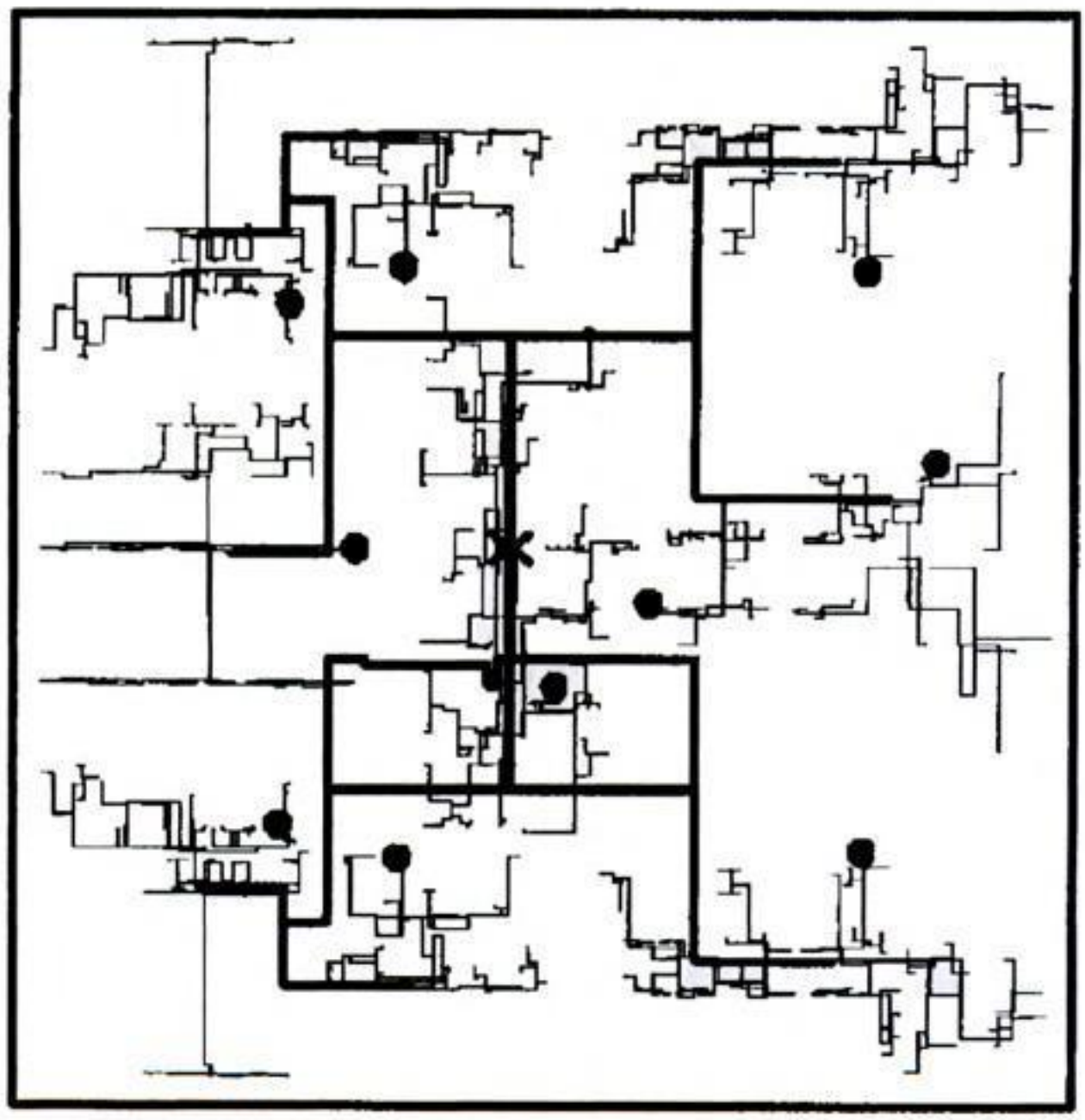


Figura 10-20 Esempio di rete di distribuzione adattata RC in un microprocessore IBM [Restle98].

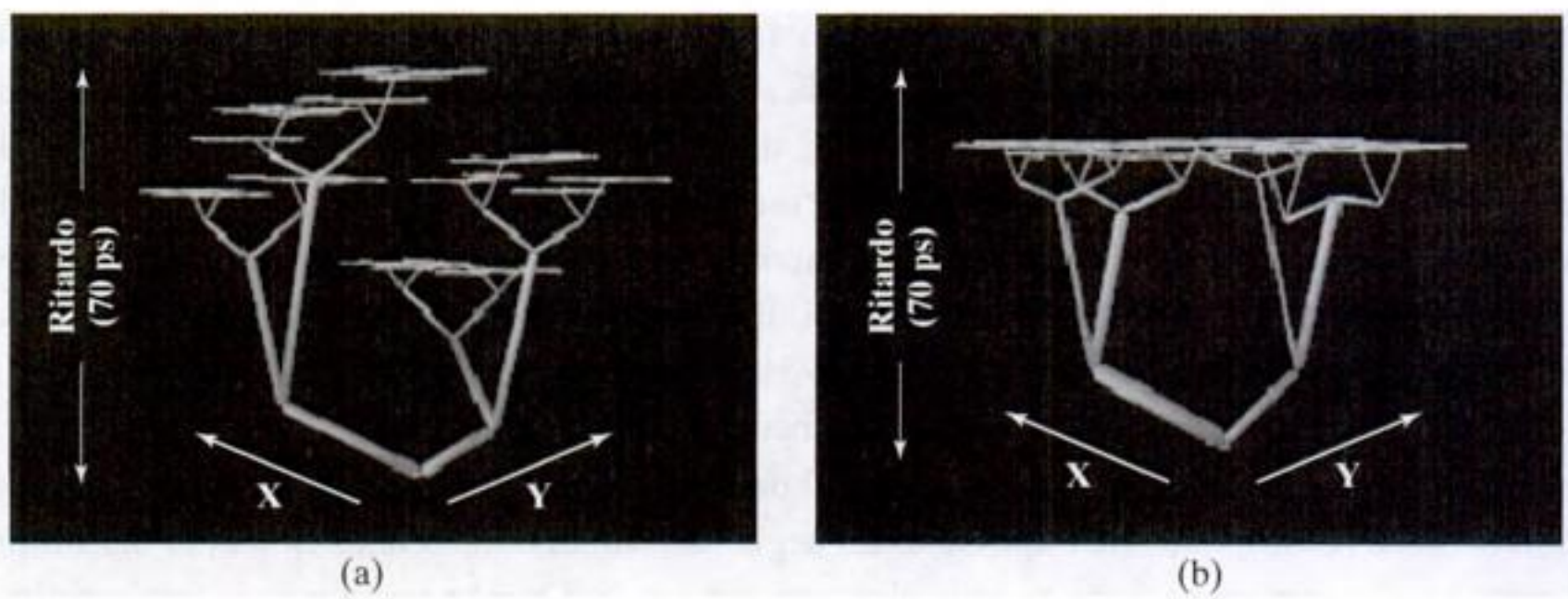


Figura 10-21 Visualizzazione del ritardo del clock in una rete ad albero che pilota carichi diversi. Gli assi *X* e *Y* rappresentano il chip, mentre l'asse *Z* rappresenta il ritardo del clock. Lo spessore delle linee è proporzionale al larghezza delle piste. Il carico sbilanciato dà origine a uno skew notevole, come è evidente in (a). Adattando con cura lo spessore delle piste, è possibile bilanciare il carico, minimizzando lo skew, come mostrato (b) [Restle01]. Vedi anche l'inserto a colori 9.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

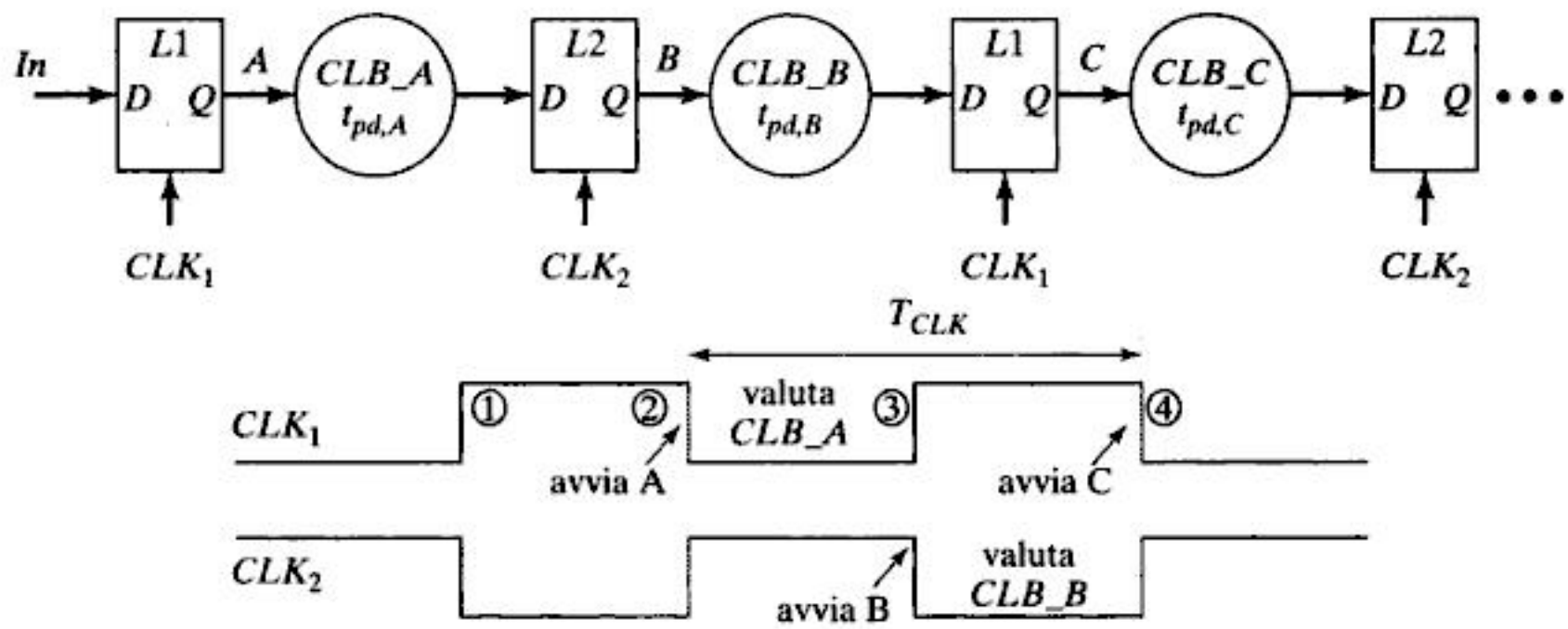


Figura 10-28 Sistema a latch nel quale i latch trasparenti sono separati da blocchi di logica combinatoria.

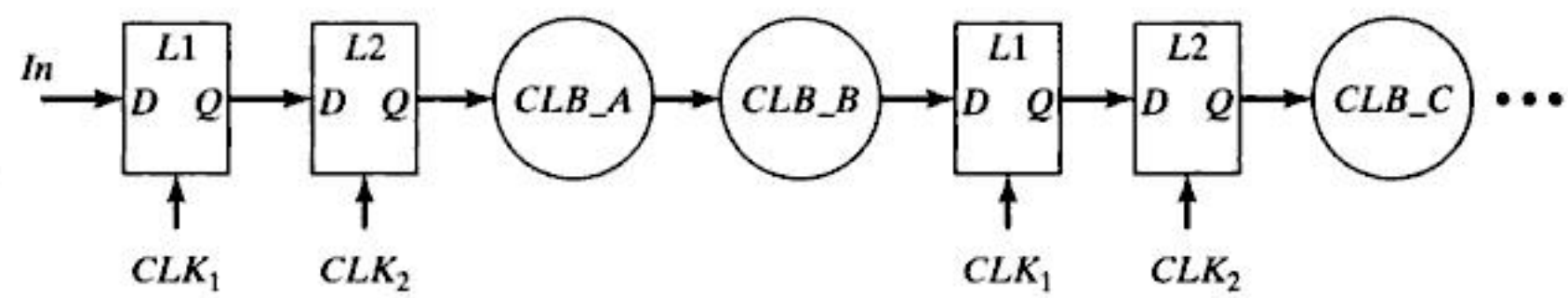


Figura 10-29 Versione pipeline edge-triggered (la cascata di due latch con clock complementari è equivalente a un registro edge-triggered) della logica della Figura 10-28.

trasferimento del tempo residuo da un blocco al successivo è automatico. Il vantaggio principale del prestito del residuo è quello di permettere ad alcuni blocchi logici di impiegare più di un ciclo di clock per la propria elaborazione, pur rispettando i vincoli temporali a livello globale. Detto in altro modo, se un sistema sequenziale lavora a una determinata frequenza di clock, e il ritardo logico totale in un ciclo completo è maggiore del periodo di clock, allora il tempo in più, o *residuo* (*slack*), è stato implicitamente preso in prestito da stadi precedenti. Da un punto di vista formale, si è verificata la propagazione del residuo se $T_{CLK} < t_{pd,A} + t_{pd,B}$ e la logica funziona correttamente (per semplicità ignoriamo i ritardi dei latch). Questo implica che la frequenza di clock può essere maggiore rispetto al vincolo del percorso critico.

Come detto in precedenza, il trasferimento del tempo residuo discende dal fatto che i latch sono sensibili al livello del clock. Nel circuito mostrato nella Figura 10-28, l'ingresso a *CLB_A* deve essere valido entro il fronte di discesa dei CLK_1 (fronte ②). Cosa succede se il blocco di logica combinatoria dello stadio precedente termina in anticipo la propria elaborazione, fornendo un dato di ingresso valido per *CLB_A* prima del fronte ②? Poiché il latch è trasparente durante tutta la fase alta del proprio clock, il dato in ingresso a *CLB_A* è valido non appena lo stadio precedente ha concluso la sua elaborazione. Questo implica che il tempo massimo a disposizione per *CLB_A* è la durata della sua fase di valutazione (cioè la fase bassa di CLK_1) più qualsiasi periodo di tempo avanzato dall'elaborazione precedente.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

loro uscita è disponibile una parola valida, cioè pronta per essere ulteriormente elaborata. Questo scambio di informazioni garantisce il corretto ordine logico degli eventi e può essere realizzato con l'ausilio di altri due segnali di controllo: *Con(ferma)* e *Ric(hiesta)*. Nel caso della struttura pipeline del nostro esempio, la sequenza potrebbe essere la seguente:

1. Arriva una parola in ingresso e viene attivato il segnale *Ric(hiesta)* del blocco *F*1. Se in quel momento *F*1 è inattivo, i dati vengono memorizzati in *R*1e il segnale *Con(ferma)* viene attivato per segnalare al buffer di ingresso che può procedere con l'acquisizione di una nuova parola di ingresso.
2. *F*1 viene abilitato attraverso il segnale *Inizia*; dopo una certa quantità di tempo, che dipende dal valore dei dati e dalle condizioni operative, l'elaborazione è completata e quindi viene attivato il segnale *Fine*.
3. A questo punto si può trasmettere un segnale *Ric(hiesta)* al modulo *F*2; se questo blocco è pronto, risponde attivando il segnale *Con(ferma)*, e quindi i dati in uscita a *F*1 vengono acquisiti e *F*1 può procedere con la sua operazione successiva.

Di fatto il sistema autotemporizzato separa le funzioni di ordinamento logico e fisico, implicite nella temporizzazione del circuito. Il segnale di completamento *Fine* assicura il rispetto dei vincoli temprali fisici cioè che il circuito sia a regime prima di accettare un nuovo ingresso. L'ordinamento logico delle operazioni è assicurato invece dallo schema *Con–Ric*, spesso chiamato *protocollo di handshaking*. I due blocchi coinvolti si sincronizzano tra loro con accordo mutuo, "stringendosi la mano" (*shaking hands*). Il protocollo di ordinamento appena descritto viene realizzato dal modulo HS ed è solo uno dei tanti possibili. La scelta del protocollo è ovviamente importante, perché ha conseguenze notevoli in termini di prestazioni e robustezza.

Nel confronto con l'approccio sincrono, i circuiti autotemporizzati mettono in evidenza alcune proprietà allettanti:

- Al contrario di quanto accade nel sistema sincrono, i segnali di temporizzazione sono tutti generati *localmente*, evitando tutti i problemi e i circuiti addizionali associati alla distribuzione di clock ad alta velocità.
- La separazione tra i meccanismi di ordinamento fisico e logico introduce un potenziale aumento delle prestazioni. Nei sistemi sincroni, il periodo del clock deve essere sufficientemente lungo da comprendere il percorso più lento, per qualsiasi sequenza in ingresso. Nei sistemi autotemporizzati, una parola in uscita da un blocco, una volta pronta, non deve attendere l'arrivo del fronte di clock successivo per procedere allo stadio di elaborazione seguente. Dato che il ritardo dei vari circuiti dipende molto spesso dal valore effettivo dei dati, un circuito autopolarizzato lavora alla *velocità media* dell'hardware, al contrario del modello sincrono che fa riferimento al *caso peggiore*. Nel caso di un sommatore ripple-carry, la profondità media della propagazione del riporto è $O(\log(N))$. Questa è una caratteristica che può essere sfruttata in un cir-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

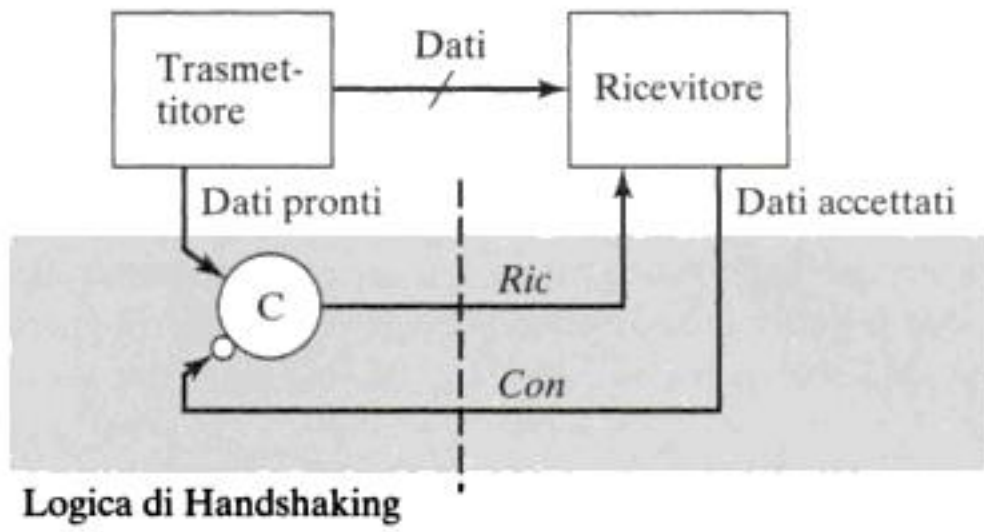


Figura 10-41 Uso di un elemento Muller *C* per realizzare un protocollo di handshaking a due fasi. Il cerchietto all'ingresso inferiore dell'elemento *C* indica un'inversione.

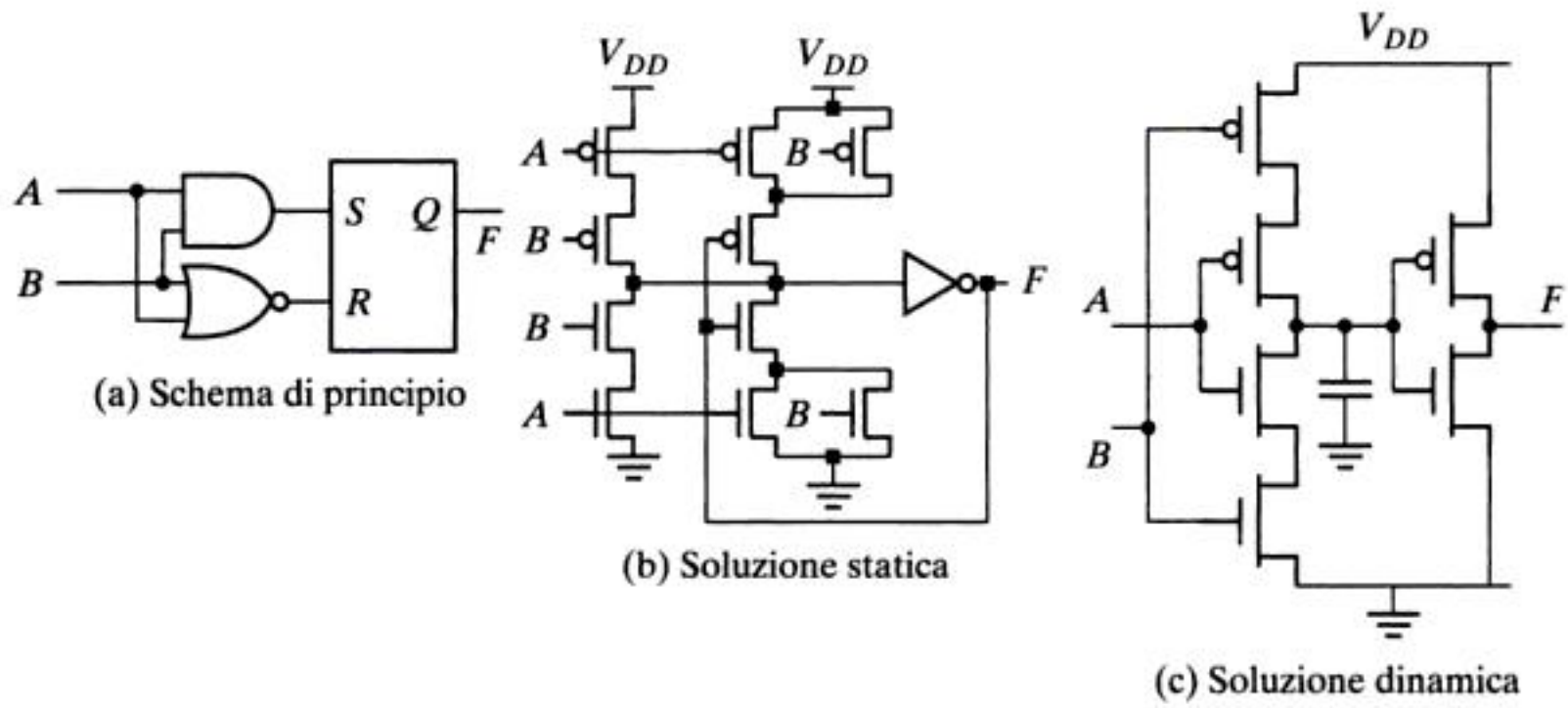


Figura 10-42 Possibili realizzazioni di un elemento Muller *C*.

data la somiglianza tra le relative tabelle di verità. La Figura 10-42 propone una realizzazione statica e una dinamica di questa funzione.

La Figura 10-41 mostra come usare questo elemento per realizzare il protocollo di handshaking a due fasi nell'esempio dei sistema trasmettitore–ricevitore. Ipotizziamo che i segnali *Ric*, *Con* e *dati pronti* siano inizialmente 0. Quando il trasmettitore vuole inviare la parola successiva, il segnale *dati pronti* è portato a 1, innescando l'*elemento C*, poiché entrambi i suoi ingressi si trovano a 1. Il segnale *Ric* si porta a livello alto (generalmente indichiamo questo evento con il simbolo *Ric*↑). Il trasmettitore ora si trova in attesa e il controllo passa al ricevitore. L'*elemento C* è bloccato e non possono essere trasmessi nuovi dati al bus (*Ric* rimane alto) finché i dati già trasmessi non vengono elaborati dal ricevitore. Non appena questo accade, il segnale *dati accettati* esegue la commutazione basso-alto; questo evento può essere il risultato di diverse azioni, che possono anche coinvolgere altri elementi *C* che comunicano con i blocchi seguenti. Comunque l'effetto che ne consegue è un evento *Con*↑, che sblocca l'*elemento C* e restituisce il controllo al trasmettitore. Un evento *dati pronti*↓, che eventualmente può essere accaduto anche prima di *Con*↑, genera *Ric*↓, e il ciclo si ripete.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

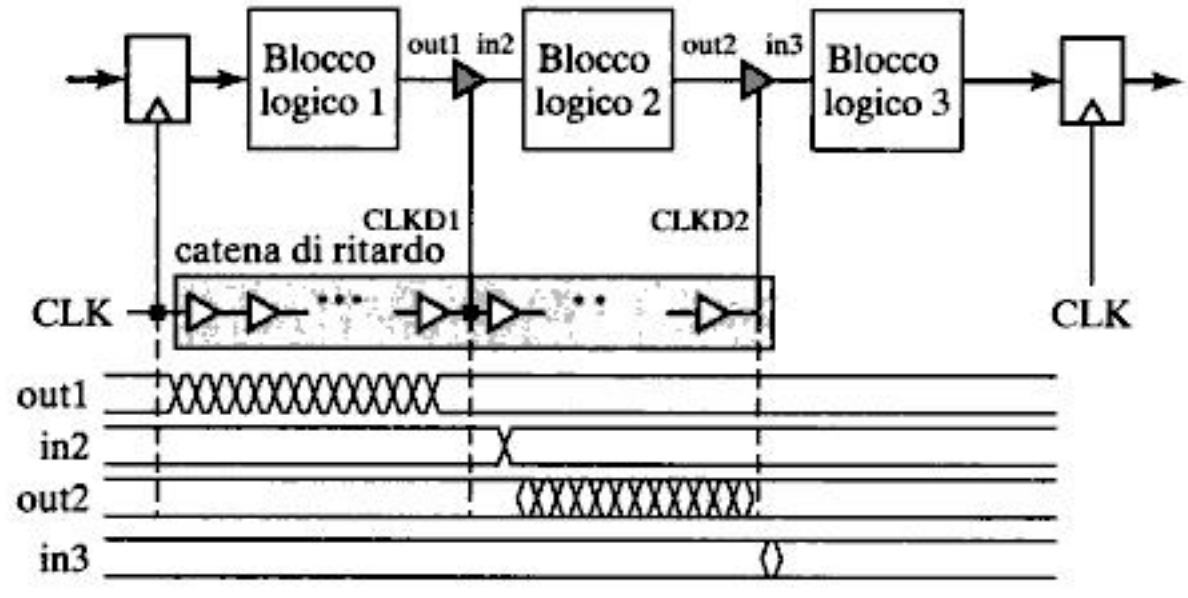


Figura 10-48 Uso dell'autotemporizzazione per ridurre i glitch.

transizioni dovute ai glitch, i buffer tristate dovrebbero essere abilitati solo quando si è sicuri che le uscite del blocco logico 1 siano stabili e valide. Il controllo dei buffer tristate può essere realizzato mediante l'uso di un segnale di abilitazione autotemporizzato, generato trasmettendo il clock di sistema attraverso una catena di ritardo che emula il percorso critico del processore: la catena di ritardo è quindi intercettata nei punti corrispondenti alla posizione dei buffer. Questa tecnica è efficace nel ridurre notevolmente la capacità effettiva dei blocchi logici combinatori, quali i moltiplicatori, persino tenendo conto dell'aggiunta della linea di ritardo, della distribuzione del segnale di controllo e dei buffer [Goodman98].

Logica a postcarica

Una forma interessante di circuiti autotemporizzati è la logica CMOS ad autoinizializzazione. Questo sistema usa una struttura di controllo diversa da quella della convenzionale pipeline autotemporizzata. Invece di aspettare che tutti gli stadi logici abbiano completato la propria elaborazione per portare il sistema nello stato di inizializzazione, l'idea è quella di porre in stato di precarica un blocco logico non appena abbia completato la sua operazione. Poiché la precarica avviene dopo l'operazione logica invece che prima della valutazione, questa tecnica è spesso indicata con il termine *logica a postcarica*. Un diagramma a blocchi della logica a postcarica è mostrato nella Figura 10-49 [Williams00]. Come si può vedere da questo diagramma, la precarica di L1 avviene quando lo stadio successivo ha completato la sua elaborazione e quindi non ha più bisogno del suo ingresso. È possibile anche precaricare un blocco direttamente in base alla sua stessa uscita, ma bisogna fare attenzione che lo stadio seguente abbia fatto in tempo a svolgere la sua elaborazione, rendendo il dato obsoleto.

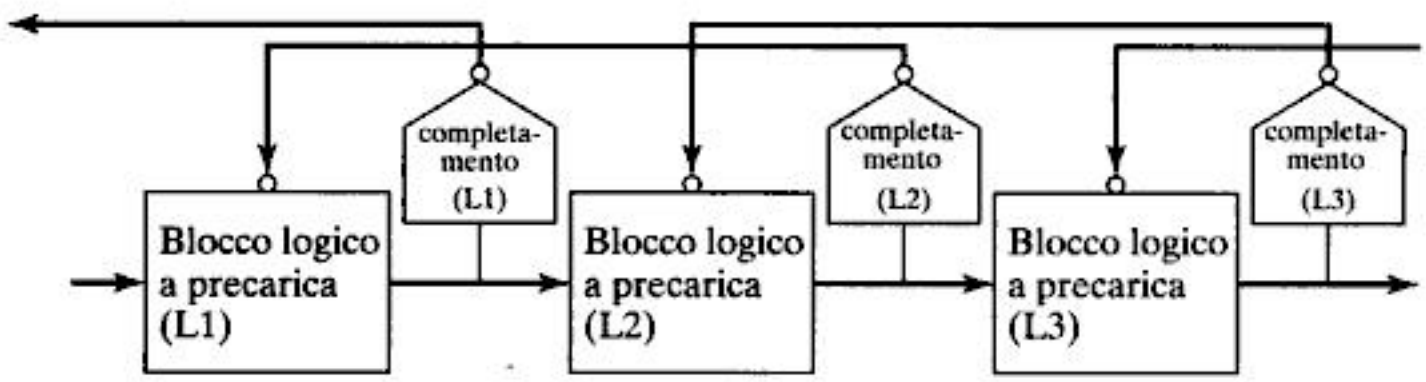


Figura 10-49 Logica ad autoinizializzazione.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

ottiene, entro certi limiti, con l'introduzione di un *filtro di anello*. Questo filtro passa-basso rimuove le componenti ad alta frequenza della tensione di controllo del VCO, smorzando la sua risposta e riducendo il jitter. Dobbiamo però ricordare che un PLL è un sistema in retroazione e quindi l'introduzione di un ulteriore sfasamento da parte del filtro può portare a instabilità. I parametri caratteristici di un PLL sono l'*intervallo di aggancio* (*lock range*), cioè l'intervallo di frequenze di ingresso per cui l'anello conserva la sua funzionalità; il *tempo di aggancio* (*lock time*), cioè il tempo necessario al PLL per agganciare un determinato segnale di ingresso; infine il jitter. Quando il sistema è agganciato, la frequenza del clock di sistema è N volte la frequenza del clock di riferimento.

Un PLL è un circuito analogico ed è inerentemente sensibile al rumore e alle interferenze. Queste considerazioni sono particolarmente applicabili al filtro di anello e al VCO, per il quale il rumore ha un effetto diretto sul jitter. Una delle sorgenti principali di interferenza è il rumore che si accoppia al circuito attraverso le linee di alimentazione e il substrato. Questo è un problema particolarmente importante nell'ambito di circuiti digitali, in cui esistono diverse sorgenti di rumore. Per questo motivo sono particolarmente indicati circuiti analogici con un'elevata reiezione della tensione di alimentazione, come i VCO differenziali [Kim90]. In definitiva, integrare un componente altamente sensibile in un ambiente digitale ostile non è banale e richiede un progettista analogico esperto. I paragrafi che seguono descrivono più dettagliatamente i vari elementi che compongono un PLL.

10.6.2 Blocchi fondamentali di un PLL

Oscillatore controllato in tensione (VCO)

Un VCO genera un segnale periodico con una frequenza che è una funzione lineare della tensione di controllo in ingresso v_{cont}. In altre parole, la frequenza del VCO può essere espressa come:

$$\omega = \omega_0 + K_{vco} \cdot v_{cont} \tag{10.13}$$

Dato che la fase è l'integrale nel tempo della frequenza, la fase di uscita del VCO è data da:

$$\phi_{out} = \omega_0 t + K_{vco} \cdot \int_{-\infty}^{t} v_{cont} dt \tag{10.14}$$

dove K_{vco} è il guadagno del VCO espresso in rad/s/V, ω_0 è una differenza di frequenza fissa e v_{cont} controlla la frequenza centrata intorno a ω_0. Il segnale di uscita ha la forma:

$$x(t) = A \cdot \cos\left(\omega_0 t + K_{vco} \cdot \int_{-\infty}^{t} v_{cont} dt\right) \tag{10.15}$$

Alcune possibili tecniche per la realizzazione di un VCO sono state presentate nel Capitolo 7; al momento gli oscillatori ad anello differenziali sono il metodo preferito.

Rilevatore di fase

Il rilevatore di fase valuta la differenza di fase relativa tra i due segnali di ingresso e fornisce in uscita un segnale che è proporzionale a tale differenza. Uno degli ingressi al rilevatore di

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

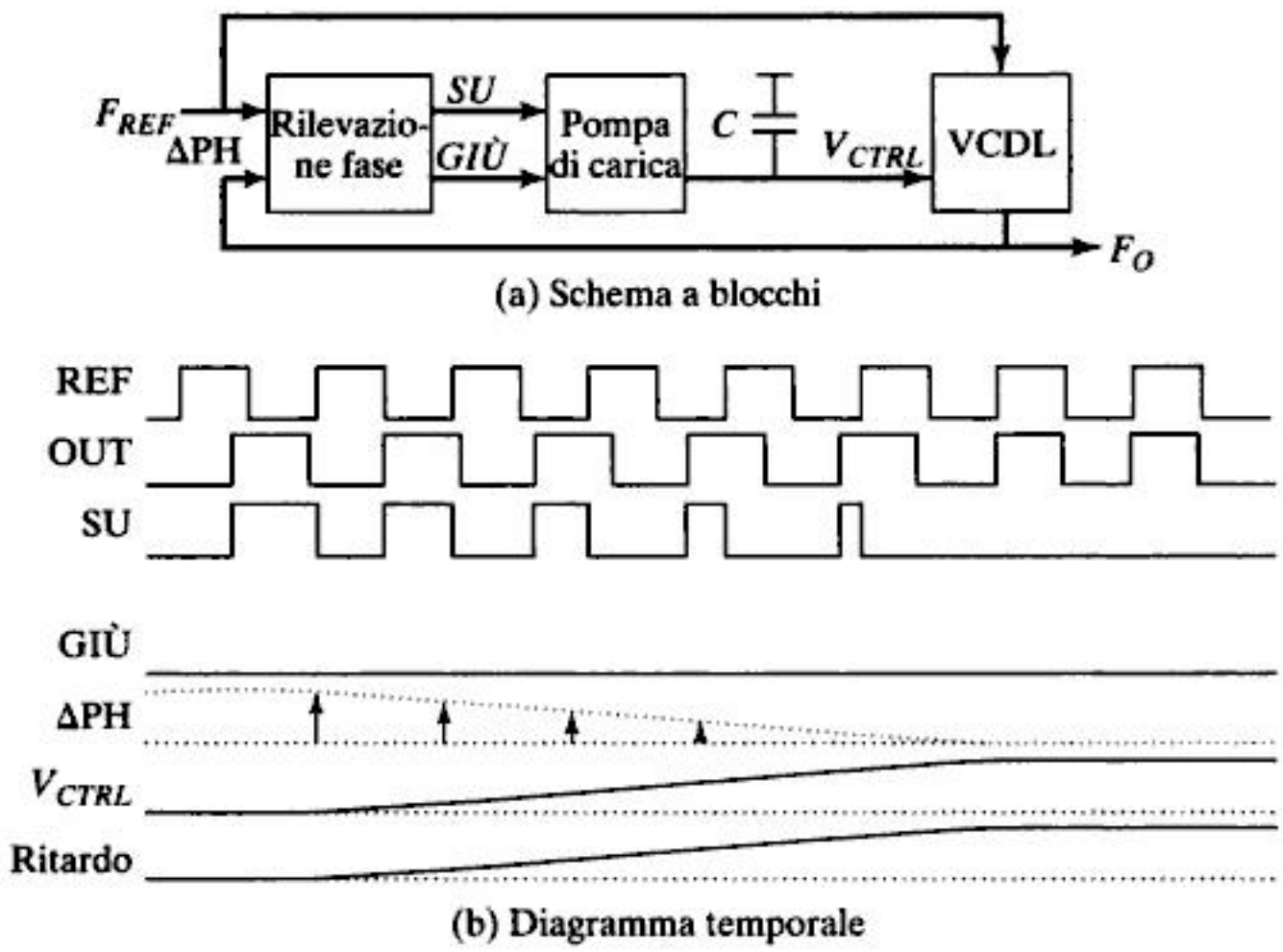


Figura 10-67 Anello ad aggancio di ritardo [Maneatis00].

10.7 Sviluppi e prospettive futuri

Questo paragrafo mette in luce alcune delle attuali tendenze per l'ottimizzazione della temporizzazione che tendono a combinare le prestazioni elevate con il consumo di potenza ridotto.

10.7.1 Clock distribuiti e DLL

Una tendenza recente nelle temporizzazioni di circuiti ad alte prestazioni è l'uso di *anelli ad aggancio di ritardo* (DLL, *delay locked loop*), una variante del PLL. Lo schema circuitale di un DLL è riportato nella Figura 10-67a [Maneatis00]. L'elemento principale di un DLL è la *linea di ritardo controllata in tensione* (VCDL, *voltage-controlled delay line*): consiste nella cascata di elementi di ritardo programmabili (ad esempio invertitori controllati in corrente). L'idea fondamentale è di **ritardare il clock di uscita in modo che si allinei perfettamente con quello di riferimento**. Diversamente dal caso del VCO, non vi è alcun generatore di clock; la frequenza di riferimento entra direttamente in ingresso al VCDL. Analogamente al caso del PLL, un rilevatore di fase confronta la frequenza di riferimento con l'uscita della linea di ritardo (F_O) e genera un segnale di errore *SU–GIÙ*. Notiamo che qui è richiesta solo una rilevazione di fase e non anche di frequenza. Quando l'anello è agganciato, non c'è errore tra i due clock. La funzione della retroazione è di regolare il ritardo lungo la VCDL in modo che il fronte di salita del clock di riferimento in ingresso (f_{REF}) e quello del clock di uscita (f_O) siano allineati.

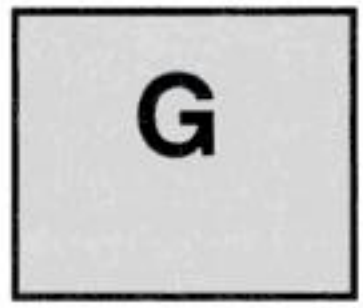
Un andamento temporale qualitativo dei segnali nel DLL è mostrato nella Figura 10-67b. Inizialmente il DLL non è agganciato; poiché il primo fronte del clock di uscita arriva prima di quello di riferimento, viene generato un impulso su *SU* di durata uguale all'errore

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.



Verifica del progetto

Simulazione e verifica

Verifica elettrica e della temporizzazione

Verifica formale

Finora ci siamo affidati spesso alla simulazione, come mezzo preferenziale per garantire la correttezza di un progetto e per estrarne i parametri critici quali la velocità e il consumo di potenza. La simulazione è però solo una delle tecniche a disposizione del progettista per raggiungere questi scopi. In linea generale possiamo distinguere tra *simulazione* e *verifica*.

Nella *simulazione*, il valore di un parametro del sistema, quale il margine di rumore, il tempo di propagazione o l'energia dissipata, viene determinato applicando un insieme di stimoli al modello scelto per il circuito ed estraendo il parametro in questione dall'andamento temporale dei segnali all'interno del circuito. Sebbene questo approccio sia molto flessibile, ha lo svantaggio che i risultati ottenuti dipendono fortemente dalla scelta degli stimoli. Per esempio, un evento di ridistribuzione della carica in una porta in logica dinamica, non è rilevato dalla simulazione se non viene applicata in ingresso l'esatta configurazione che causa questo fenomeno. Analogamente, il ritardo di un sommatore varia molto a seconda dei dati in ingresso. L'identificazione del ritardo al caso peggiore richiede una scelta oculata dei vettori di stimolo, in modo da attivare l'intero percorso di propagazione del riporto. In altre parole, il progettista deve avere una buona comprensione delle caratteristiche intrinseche del modulo circuitale e del suo funzionamento. In mancanza di tali fattori si possono ottenere risultati senza senso.

Materiale protetto da copyright

La *verifica*, d'altra parte, cerca di estrarre i parametri del sistema direttamente dalla descrizione circuitale. Per esempio, il percorso critico di un sommatore può essere riconosciuto dall'osservazione diretta dello schema circuitale o di un suo modello. Questo approccio fa sì che il risultato sia indipendente dalla scelta dei vettori di stimolo ed è potenzialmente infallibile. D'altra parte, si basa su un certo numero di ipotesi implicite rispetto alle tecniche e alle metodologie di progetto. Nel caso del sommatore, la determinazione del tempo di propagazione richiede una comprensione delle operazioni logiche dei vari componenti circuitali (per esempio, logica statica o dinamica) e la definizione del termine *tempo di propagazione*. Come secondo esempio, è necessario identificare gli elementi che realizzano i registri prima di poter determinare la massima velocità di clock di un sistema sincrono. Gli strumenti di verifica hanno quindi uno scopo limitato e gestiscono solo una classe limitata di stili circuitali, quali i sistemi sincroni a fase singola. In questo inserto analizzeremo un certo numero di tecniche di verifica comuni.

Verifica elettrica

Dato lo schema circuitale a livello di transistor di un sistema digitale, è possibile verificare se un certo numero di regole fondamentali è rispettato o meno. Alcuni esempi di queste regole sono di aiuto per illustrare meglio questo concetto, e molti altri possono essere ricavati dalle considerazioni fatte nei capitoli precedenti.

- Il numero di inversioni tra due porte C^2MOS deve essere pari.
- In una porta pseudo-NMOS il rapporto fra le dimensioni del transistor PMOS di pull-up e quelle dei transistor NMOS di pull-down deve rispettare limiti precisi, al fine di garantire un sufficiente margine al rumore per il livello basso NM_L.
- Si può imporre un limite minimo alle dimensioni dei transistor in funzione del fan-out, per garantire che i tempi di salita e discesa restino entro certi limiti.
- Il valore massimo di ridistribuzione della carica in un sistema dinamico deve essere tale da non violare il vincolo imposto dal margine al rumore alto.

Il buon senso può aiutare a definire un ampio insieme di regole che un sistema dovrebbe sempre rispettare. Applicarle richiede però una profonda conoscenza della struttura circuitale. Uno strumento di verifica elettrica quindi, comincia con l'identificazione di sottostrutture note, all'interno del circuito completo. Dei modelli tipici sono porte logiche fondamentali, pass transistor e registri. Lo strumento verifica sulla rete equivalente regola dopo regola. Dato che le regole elettriche tendono a essere specifiche per un particolare stile, dovrebbero essere facilmente modificabili. Per esempio *sistemi esperti basati su regole* prevedono facili aggiornamenti della base di regole [DeMan85]. Le singole regole possono anche essere complesse e richiedere addirittura la simulazione di piccole sezioni della rete per verificare se una data condizione è soddisfatta. In definitiva, la verifica elettrica è uno strumento utile e può ridurre significativamente il rischio di malfunzionamenti.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Esempio G.1 Esempio di verifica temporale

L'uscita di uno strumento di verifica temporale, PathMill della Synopsys [PathMill], è mostrata nella Figura G-2. L'ingresso al processo di verifica è la netlist di transistor; si possono includere anche modelli a livello di porte logiche e blocchi. L'analisi considera gli elementi parassiti capacitivi e resistivi che sono ottenuti dallo schema a livello di transistor o da estrazione dal layout.

L'uscita dell'analisi temporale è una lista ordinata di percorsi critici. Per l'esempio della Figura G-2, il percorso più lungo si estende dal nodo B (fronte di discesa) al nodo Y (fronte di discesa) attraverso i nodi S1, S2 e X3. Il ritardo previsto è di 1.14 ns. Altri percorsi vicino a quello critico sono in ordine decrescente di ritardo: $B\,(F) \rightarrow Y\,(R)$ (1.11 ns), $B\,(R) \rightarrow Y\,(F)$ (1.04 ns), $C\,(R) \rightarrow Z\,(R)$ (1.04 ns) e $D\,(R) \rightarrow Y\,(R)$ (1.00 ns). Osserviamo che il circuito contiene diversi pass transistor in cascata che rendono l'analisi complicata e aumentano la probabilità di individuare un percorso falso. Un esempio di questo tipo è il percorso $A\,(F) \rightarrow X1 \rightarrow X3 \rightarrow Z(R)$ per $B = 0$. Per un valore basso di B, il nodo Z in realtà può solo effettuare una commutazione a livello basso.

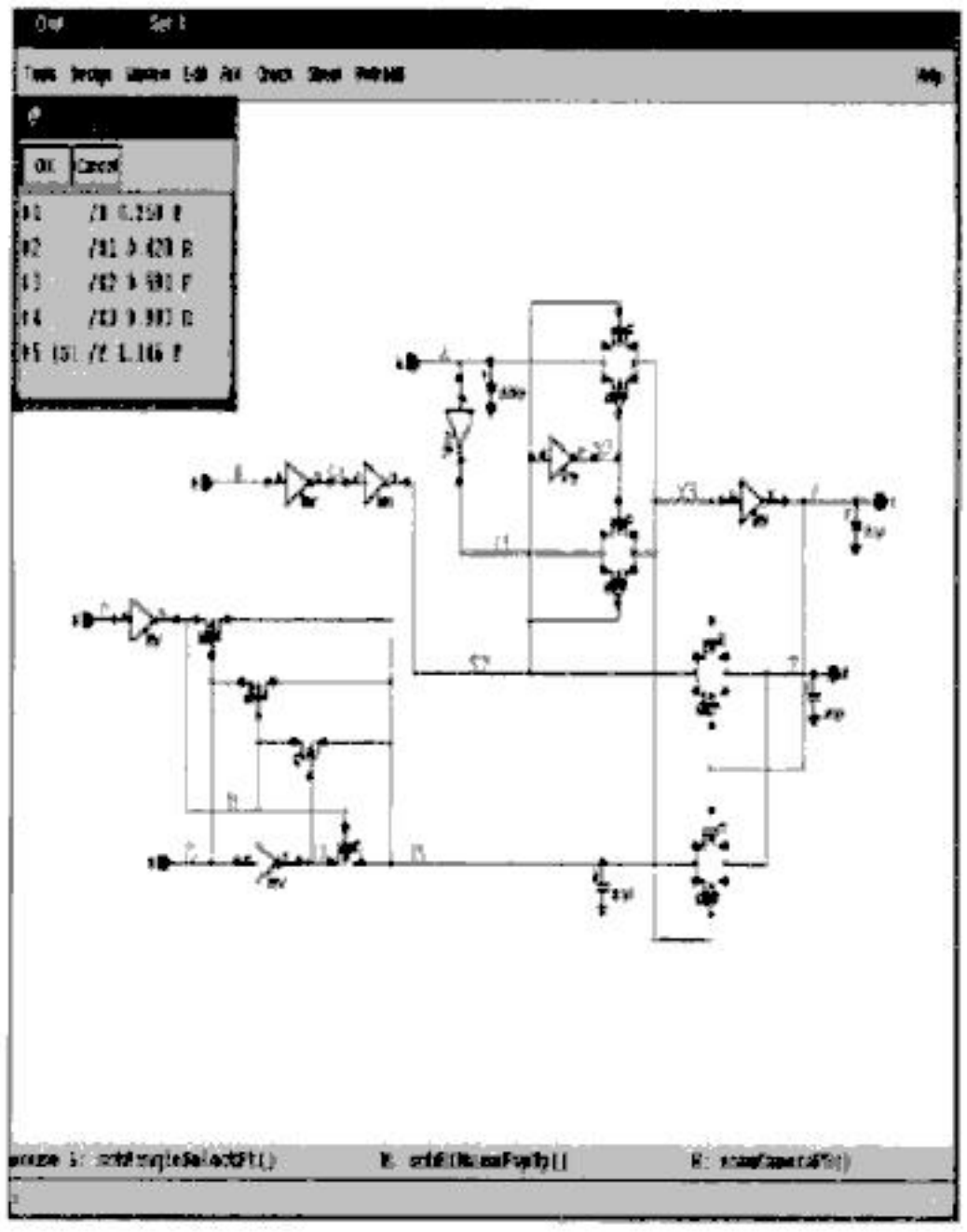


Figura G-2 Esempio della risposta di uno strumento di verifica statico, PathMill (*gentile concessione di Synopsys, Inc.*). I risultati sono visualizzati usando *Cadence DFII*. Il percorso critico si estende dal nodo *B* al nodo Y attraverso i nodi *S1*, *S2* e *X3*.

Materiale protetto da copyright

CAPITOLO

11

Blocchi logici e aritmetici

Progettare sommatori, moltiplicatori e traslatori per ottimizzare le prestazioni, l'area e il consumo di potenza

Ottimizzazione a livello logico e di sistema

Compromesso tra consumo e velocità

11.1 Introduzione
11.2 Architettura dell'unità logico-aritmetica dei microprocessori
11.3 I sommatori
- 11.3.1 Definizioni generali
- 11.3.2 Ottimizzare del circuito full-adder
- 11.3.3 Ottimizzare l'architettura di un sommatore

11.4 I moltiplicatori
- 11.4.1 L'operazione di moltiplicazione binaria
- 11.4.2 Calcolo dei prodotti parziali
- 11.4.3 Accumulazione dei prodotti parziali
- 11.4.4 L'addizione finale

11.5 I traslatori
- 11.5.1 Traslatore barrel
- 11.5.2 Traslatore logaritmico

11.6 Altri operatori aritmetici
11.7 Compromesso tra velocità e consumo nelle unità di elaborazione *
- 11.7.1 Tecniche di riduzione del consumo in fase di progetto
- 11.7.2 Gestione del consumo durante il funzionamento
- 11.7.3 Riduzione del consumo statico in fase di inattività (standby)

11.8 Prospettive: scelte e compromessi nella progettazione

Materiale protetto da copyright

11.9 Riepilogo
11.10 Approfondimenti

11.1 Introduzione

Dopo aver studiato approfonditamente la progettazione e l'ottimizzazione dei principali circuiti digitali, è tempo di guardare verso un livello di astrazione superiore, usando ciò che abbiamo imparato in un'ottica orientata maggiormente al sistema. Nel seguito, applicheremo tutte le tecniche apprese nei capitoli precedenti alla progettazione di alcuni circuiti digitali spesso presenti nelle unità logico-aritmetiche dei microprocessori e dei DSP. Più in dettaglio, analizzeremo la progettazione dei moduli più rappresentativi di un microprocessore, come i sommatori, i moltiplicatori e i traslatori (*shifter*). Molto spesso la velocità e il consumo di questi elementi domina sulle prestazioni globali del sistema, richiedendone quindi un accurata ottimizzazione. Deve essere chiaro fin dall'inizio che il progetto di questi blocchi non è un compito facile, dato che per ciascuno di essi esistono molteplici strutture logiche e diverse realizzazioni circuitali, con pregi e difetti in termini di area, velocità e consumo di potenza.

Sebbene lontana dall'essere esaustiva, la trattazione seguente aiuta a mettere a fuoco i compromessi essenziali nel corso della progettazione di un sistema digitale complesso. Vedremo che ottimizzare il sistema sulla base di un singolo livello di astrazione (ad esempio, il dimensionamento dei transistor) conduce a prestazioni globali inferiori del prodotto finale. I progettisti devono avere dunque una visione globale, identificando quelle porte logiche, quei circuiti o quei transistor che sono critici per il traguardo da raggiungere. Le parti meno critiche del circuito possono essere sviluppate secondo la prassi comune. Svilupperemo anche modelli del primo ordine per le prestazioni di un circuito per agevolare la comprensione del suo comportamento. Infine, cercheremo di mettere in luce quali strumenti CAD possono aiutare a semplificare questa fase di progettazione.

Prima di considerare la progettazione dei moduli aritmetici, è opportuno fare un breve quadro generale del ruolo delle *unità di elaborazione (datapath)* in un processore digitale, non solo allo scopo di mettere in luce i requisiti di queste unità, ma anche per mettere il resto del capitolo nella stessa prospettiva. Gli altri moduli dei processori (I/O, controllori e memorie) hanno requisiti diversi e sono stati discussi nel Capitolo 8. Dopo che avremo analizzato il compromesso tra area, potenza e velocità nel progetto dei blocchi sommatori, moltiplicatori e traslatori, useremo queste stesse strutture per illustrare alcuni delle tecniche introdotte nel Capitolo 6 per ridurre il consumo di potenza. Il capitolo si conclude con una breve panoramica sulla progettazione delle unità di elaborazione e sui compromessi necessari.

11.2 Architettura dell'unità logico-aritmetica dei microprocessori

Nel Capitolo 8 abbiamo introdotto il concetto di processore digitale. Questo è composto da vari blocchi: le unità di elaborazione dei dati, la memoria, i moduli di controllo e i moduli di ingresso-uscita (I/O). L'unità di elaborazione è il cuore del processore, cioè il luogo dove sono eseguite tutte le operazioni logiche e aritmetiche sui dati. Tutti gli altri blocchi del processore sono unità di supporto che o immagazzinano i risultati delle operazioni dell'unità di

Materiale protetto da copyright

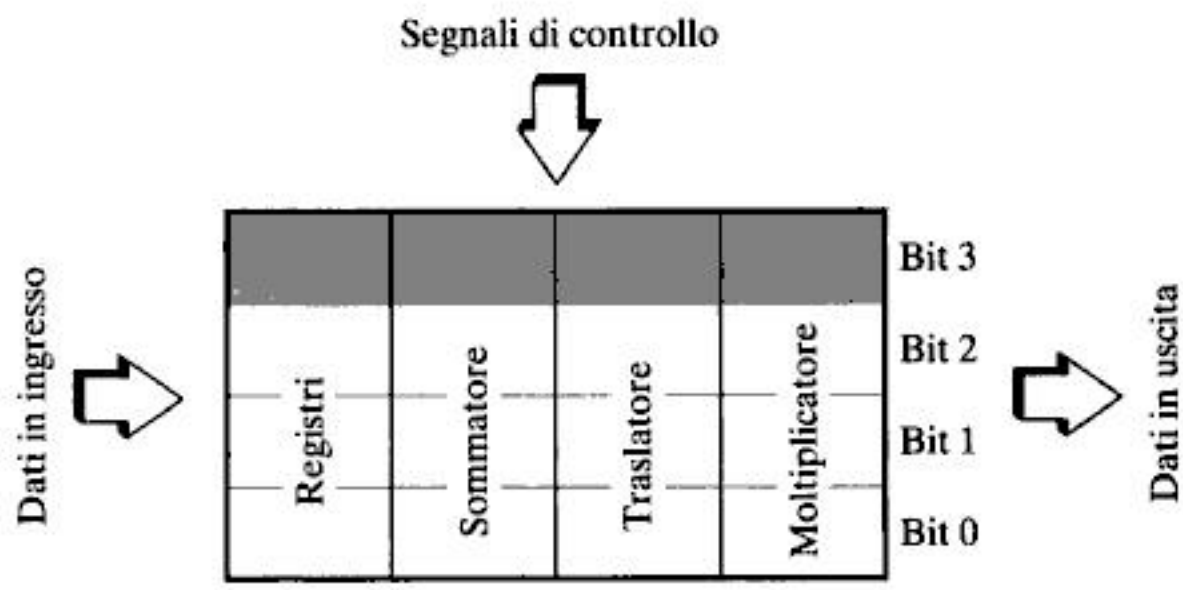


Figura 11-1 Organizzazione di un'*unità di elaborazione*.

elaborazione o controllano il flusso delle operazioni, per esempio determinando cosa deve essere fatto nel ciclo successivo. Una tipica unità di elaborazione è formata da diversi blocchi funzionali elementari interconnessi tra loro; tra questi vi sono gli operatori aritmetici (addizione, moltiplicazione, comparazione e traslazione) e gli operatori logici (AND, OR e XOR). Il progetto degli operatori aritmetici è l'argomento centrale di questo capitolo. L'applicazione finale per cui il sistema è concepito pone vincoli specifici al progetto: in alcuni casi, come i personal computer, la velocità computazionale è tutto; in molte altre applicazioni, invece, esiste un limite superiore al consumo di potenza, ovvero alla massima quantità di calore dissipabile, oppure esiste una quantità finita di energia a disposizione (ad esempio una batteria) per garantire un certo flusso di informazioni.

Generalmente un'*unità di elaborazione* è organizzata come nella Figura 11-1: invece di operare su singoli bit, i dati sono organizzati in *parole (word)*. Le unità di elaborazione degli attuali microprocessori ad uso generale sono tipicamente organizzate in parole da 32 o 64 bit, sebbene le stesse unità in processori dedicati, come quelli nei modem DSL, lettori di dischi magnetici o lettori di compact-disk siano di lunghezza arbitraria, tipicamente da 5 a 24 bit. Ad esempio, un processore a 32-bit esegue le operazioni su parole lunghe 32 bit e ciò si riflette nell'organizzazione della sua stessa unità di elaborazione. Dovendo eseguire la stessa operazione frequentemente su ciascun bit della parola, l'unità computazionale è formata da un gruppo di 32 linee di bit, ciascuna delle quali esegue la stessa operazione su singolo bit. All'interno di ciascun gruppo la stessa struttura viene replicata per ciascuna linea e quindi nel progettare l'unità di elaborazione ci si può concentrare sulla progettazione di una sola di queste linee per poi replicare la stessa struttura (o quasi) 32 volte.

11.3 I sommatori

L'addizione è l'operazione aritmetica usata più frequentemente e costituisce spesso il fattore limitante per la velocità computazionale di un sistema. Dunque, l'ottimizzazione accurata dei blocchi sommatori riveste il ruolo di massima importanza nel progetto dell'intero sistema. L'ottimizzazione procede sia a livello logico sia a livello circuitale. A livello logico è possibile riorganizzare la struttura delle varie relazioni booleane in modo tale da ottenere un circuito più piccolo e veloce. Un esempio di ottimizzazione logica è rappresentato dal

Materiale protetto da copyright

sommatore *carry lookahead* trattato più avanti in questo capitolo. D'altra parte, a livello di circuito è possibile manipolare le dimensioni dei transistor e la topologia delle porte logiche in modo tale da ottimizzare ciascun elemento che compone il sommatore. Prima di prendere in considerazione l'ottimizzazione logica e circuitale, dobbiamo introdurre le definizioni e le proprietà fondamentali di un circuito sommatore (come viene fatto in qualsiasi libro che tratti la progettazione digitale, ad esempio [Katz94]).

11.3.1 Definizioni generali

La Tabella 11-1 mostra la tabella di verità di un sommatore binario (*full adder*): A e B sono i due bit da sommare, C_i è il riporto in ingresso, S rappresenta il risultato della somma e C_o è il riporto di uscita risultante dalla somma $A + B$. Le relazioni booleane che legano gli ingressi A, B e C_i alle uscite S e C_o si possono esprimere come segue:

$$\begin{aligned} S &= A \oplus B \oplus C_i \\ &= A\bar{B}\bar{C}_i + \bar{A}B\bar{C}_i + \bar{A}\bar{B}C_i + ABC_i \\ C_o &= AB + BC_i + AC_i \end{aligned} \tag{11.1}$$

Molto spesso è utile dal punto di vista dell'implementazione definire i segnali di somma e riporto di uscita S e C_o in funzione di tre segnali intermedi: G (*Generate*), D (*Delete*) e P (*Propagate*).[1] $G = 1$ significa che il riporto in uscita (C_o) è *generato*, cioè posto a 1 indipendentemente dal valore di C_i; ($D = 1$) rappresenta il caso opposto in cui C_o è posto a 0 indipendentemente da C_i; $P = 1$ significa che il valore del riporto di uscita sarà uguale a quello di ingresso, cioè che il valore del riporto all'ingresso C_i viene *propagato* verso l'uscita C_o. Dando un semplice sguardo alla tabella di verità, possiamo ricavare le espressioni booleane che legano i segnali G, D e P ai valori logici dei bit in ingresso A e B:

$$\begin{aligned} G &= AB \\ D &= \bar{A}\bar{B} \\ P &= A \oplus B \end{aligned} \tag{11.2}$$

Da queste relazioni possiamo ricavare S e C_o in funzione di P e G (o D):

$$\begin{aligned} C_o(G, P) &= G + PC_i \\ S(G, P) &= P \oplus C_i \end{aligned} \tag{11.3}$$

Notiamo che i segnali G e P sono funzione dei soli ingressi A e B, ma non del segnale di riporto C_i. In modo analogo possiamo ricavare delle espressioni booleane per $S(D, P)$ e $C_o(D, P)$.

Un sommatore a N bit può essere realizzato nel modo più semplice mettendo in cascata N blocchi full-adder (FA), connettendo l'uscita del riporto del blocco $k-1$ ($C_{o,k-1}$) al riporto

[1] Notiamo che il segnale di propagazione (P) può essere definito anche dalla funzione OR tra A e B. In questo caso il riporto all'ingresso è propagato fino all'uscita anche quando $A = B = 1$. Nel resto del capitolo assumeremo che il segnale P sia calcolato dalla funzione XOR di A e B, salvo diversa indicazione.

Materiale protetto da copyright

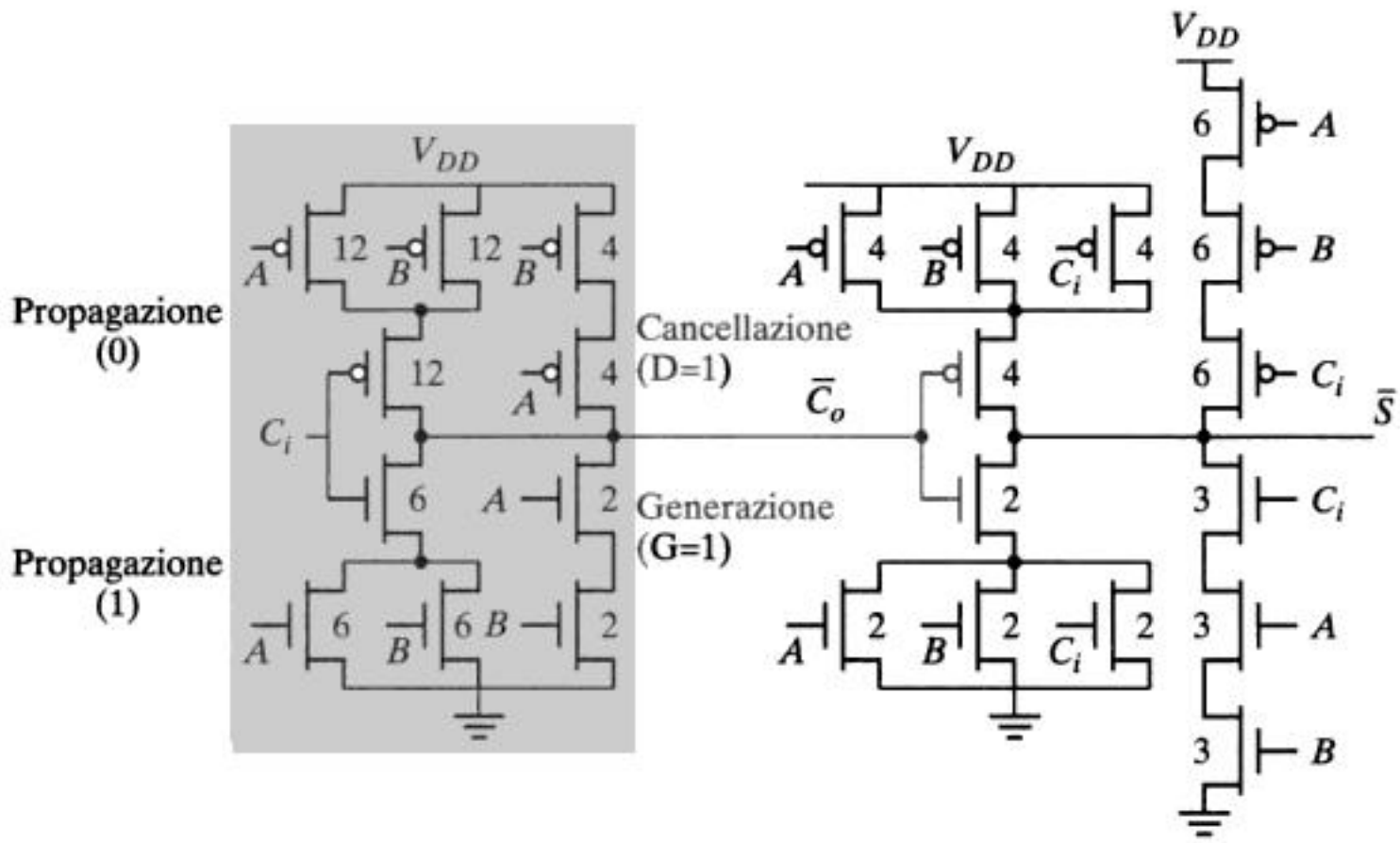


Figura 11-6 Sommatore a specchio

vi sono al massimo due transistor in serie in entrambe le reti di PDN e PUN del circuito di generazione del riporto.

- I transistor connessi a C_i sono quelli più vicini al nodo di uscita della porta logica.
- Solo i transistor che realizzano lo stadio di generazione del riporto devono essere ottimizzati per ottenere la massima velocità, mentre tutti gli altri che compongono il secondo stadio della somma possono essere lasciati di dimensione minima. Nel disegnare il layout della cella, l'obiettivo principale è quello di minimizzare la capacità del nodo $\overline{C}_o$. A tale scopo, può essere utile condividere le giunzioni dei transistor, quando è possibile, in modo da diminuire la capacità complessiva dei nodi.
- Nella cella della Figura 11-4, gli invertitori possono essere dimensionati indipendentemente per pilotare l'ingresso C_i della cella seguente. Se il circuito di generazione del riporto nella Figura 11-6 è dimensionato in modo simmetrico, ciascuno dei suoi ingressi ha un lavoro logico pari a 2. Ciò indica che il valore ottimo del fan-out (scelto per avere le massime prestazioni) dovrebbe essere (4/2) = 2. Tuttavia, l'uscita di questo stadio pilota due capacità di gate interne e sei capacità di gate della cella successiva. Una buona soluzione per mantenere tutti gli stadi uguali tra loro consiste nell'aumentare le dimensioni dei transistor nel percorso del riporto rispetto a quelli del circuito di generazione della somma di tre o quattro volte. In questo modo si ottiene il valore ottimo di fan-out (circa 2). Le dimensioni finali dei transistor sono annotate direttamente sul circuito della Figura 11-6, dove si è assunto che il rapporto PMOS/NMOS sia 2.

Sommatore a gate di trasmissione

Un full-adder può essere costruito usando multiplexer e porte logiche XOR. Sebbene questo approccio non sia facilmente praticabile in logica complementare, diventa abbastanza semplice usando la logica a gate di trasmissione. Una possibile implementazione di un full-adder

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

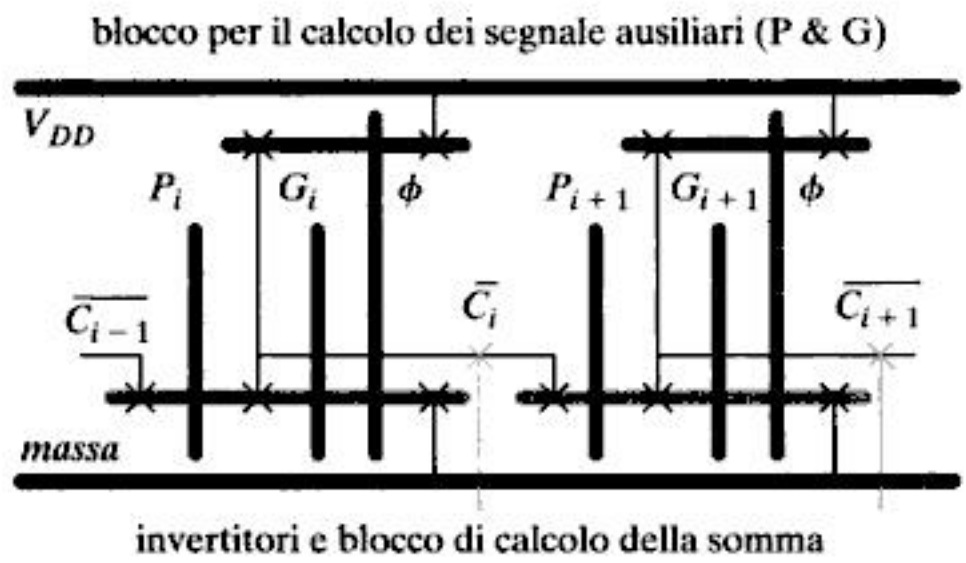


Figura 11-10 Layout semplificato di un modulo a due bit di un circuito *Manchester carry chain*.

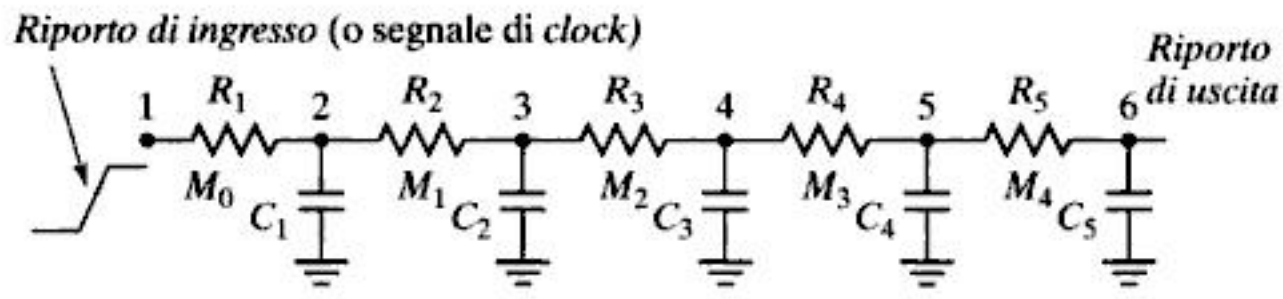


Figura 11-11 Rete *RC* equivalente per il calcolo del tempo di ritardo lungo la catena di propagazione del riporto.

Nel sommatore della Figura 11-9 il cammino critico per il calcolo del riporto (caso peggiore) può essere modellato con la rete *RC* della Figura 11-11. Come abbiamo ricavato nel Capitolo 4, il tempo di ritardo di questa catena è dato dalla seguente relazione:

$$t_p = 0{,}69 \sum_{i=1}^{N} C_i \left(\sum_{j=1}^{i} R_j \right) = 0{,}69 \frac{N(N+1)}{2} RC \tag{11.7}$$

dove, per semplicità, abbiamo ipotizzato che tutte le capacità e tutte le resistenze siano rispettivamente $C_i = C$ e $R_j = R$.

Esempio 11.2 Dimensionamento del circuito *Manchester carry chain*

Le capacità di ciascun nodo della catena sono date dalla somma di quattro capacità di drain/source, una capacità di ingresso dell'invertitore e le capacità parassite delle interconnessioni (proporzionali alla dimensione della cella). Gli invertitori e il transistor PMOS di precarica possono essere mantenuti con dimensioni minime. Tutti i contributi capacitivi fissi, incluse le capacità delle piste, possono essere stimati di 15 fF (per questa tecnologia). Se un transistor di dimensione unitaria W_0 ha una resistenza di 10 kΩ e una capacità di drain/source di 2 fF, allora la costante *RC* della rete di transistor con larghezza *W* è:

$$RC = \left(6\ \text{fF} \cdot \frac{W}{W_0} + 15\ \text{fF} \right) \cdot 10\ \text{k}\Omega \cdot \frac{W_0}{W}$$

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Calcoliamo ora il tempo di ritardo di un sommatore in architettura *carry-bypass* a N bit. Inizialmente assumiamo che il sommatore sia diviso in (N/M) blocchi di uguale lunghezza, ciascuno dei quali contiene M bit e un circuito di *bypass* che permette (condizionatamente) di trasmettere il riporto direttamente all'uscita. Un esempio è illustrato nella Figura 11-14. Una stima approssimata per il tempo di propagazione può essere derivata con l'aiuto della Figura 11-14:

$$t_{add} = t_{setup} + Mt_C + \left(\frac{N}{M} - 1\right)t_{bypass} + (M-1)t_C + t_S \tag{11.9}$$

I tempi utilizzati nell'Eq. (11.9) sono definiti come segue:

- t_{setup}: il tempo (fisso) necessario a creare i segnali ausiliari (*generazione, propagazione, cancellazione* e *bypass*).
- t_C: il tempo di ritardo del riporto per un singolo bit. Il tempo di propagazione nel caso peggiore attraverso un blocco di M bit è approssimato con M volte questo tempo.
- t_{bypass}: il tempo di propagazione attraverso il multiplexer di un singolo stadio.
- t_S: il tempo necessario a generare la somma nello stadio finale.

Nello schema a blocchi riportato nella Figura 11-14 il cammino critico è ombreggiato in grigio. Dall'Eq. (11.9) vediamo che t_p è ancora una funzione lineare del numero N di bit, dato che nel caso peggiore il riporto è comunque generato nel primo bit, si propaga attraverso tutto il primo blocco, salta i successivi ($N/M - 2$) stadi, passando per il cammino di bypass e infine si propaga attraverso l'ultimo blocco per essere utilizzato nel calcolo della somma dell'ultimo bit. Il numero ottimale di bit per ciascun blocco è funzione di svariati parametri tecnologici, tra i quali il tempo di ritardo introdotto dai multiplexer, gli eventuali buffer richiesti lungo la catena del riporto e il rapporto tra i tempi di propagazione lungo la catena e lungo il cammino di *bypass*.

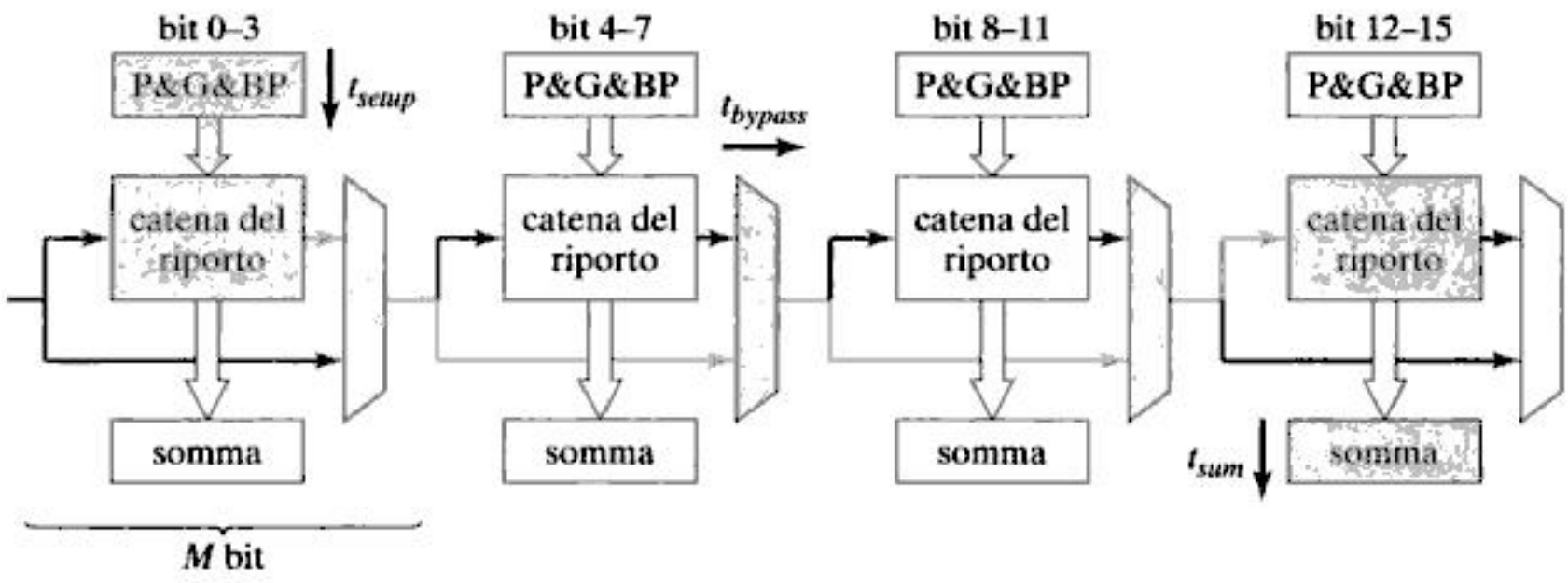


Figura 11-14 Schema a blocchi che illustra la struttura di un sommatore *carry-bypass* con 16 bit. I bit sono raggruppati in 4 blocchi con 4 bit ciascuno ($N = 16$, $M = 4$). Il cammino critico che dà luogo al caso peggiore per il tempo di propagazione è ombreggiato nello schema del sommatore.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

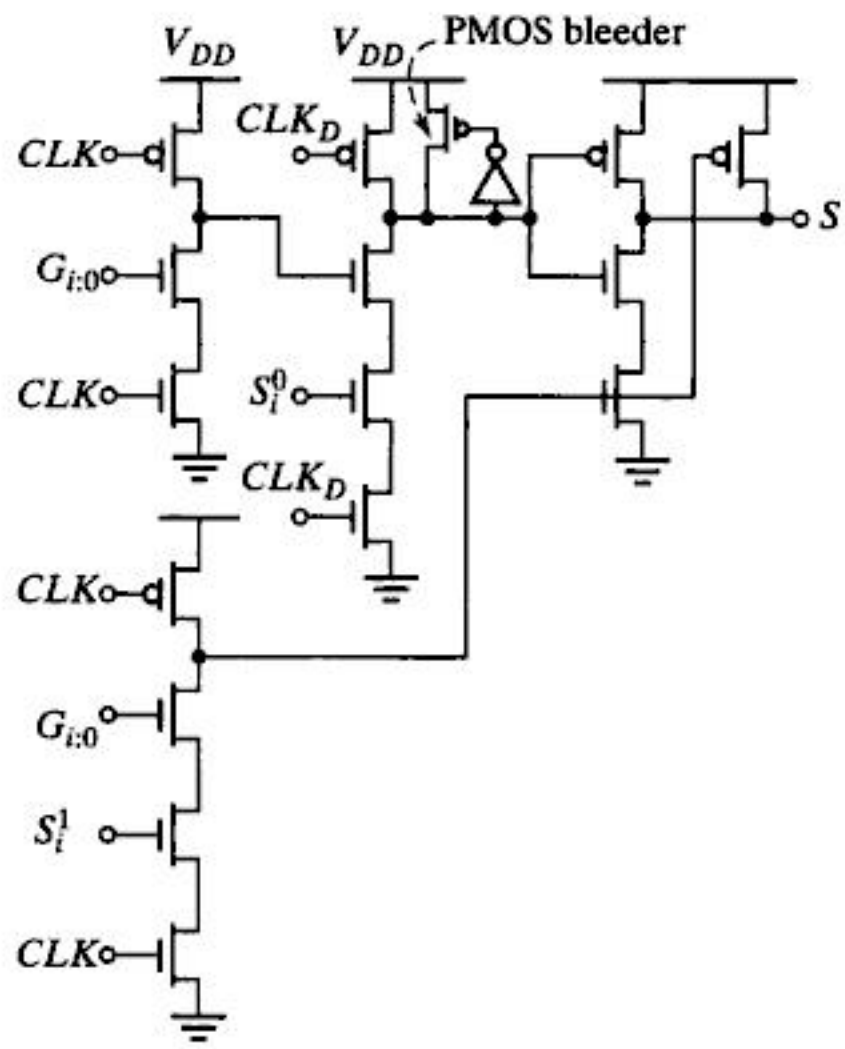


Figura 11-25 Circuito selettore del bit di somma in logica dinamica.

Vale la pena di ribadire che, quando un progettista decide di percorrere questa strada, è necessario garantire che la robustezza del circuito non venga compromessa. Tutti i segnali di clock sono soggetti a un disallineamento casuale, e quindi tutti i segnali di clock ritardati devono avere un margine di sicurezza che tenga conto del massimo disallineamento del clock. Se il fronte di salita di CLK_D arrivasse troppo presto, il sommatore potrebbe non valutare correttamente i dati in ingresso. Comunque, un dimensionamento opportuno del circuito di ritenuta può fornire una sorta di contromisura di sicurezza per il corretto funzionamento del circuito: esso infatti può eliminare il glitch all'uscita del secondo stadio, nel caso in cui il fronte del clock arrivi in anticipo. Il PMOS *bleeder* deve essere abbastanza grande per ridurre l'ampiezza del glitch, ma non tanto da compromettere le prestazioni della porta logica.

Un'alternativa è quella di implementare l'intero sommatore in logica domino differenziale, nella quale sono a disposizione sia il segnale diretto sia il valore negato, eliminando il problema dell'inversione del segnale. Tuttavia, il circuito complessivo avrebbe un consumo di potenza superiore. ■

Problema 11.4 Progetto di un sommatore in logica statica

Progettare un sommatore *carry-lookahead* a 16 bit in logica CMOS statica complementare. Realizzare l'albero di Kogge-Stone in logica negata, evitando l'impiego di invertitori addizionali. Identificare il cammino critico del sommatore. Qual è il lavoro logico delle porte lungo il cammino critico? Come possono essere dimensionate per ottenere il tempo di ritardo minimo?

Realizzazioni alternative del sommatore *lookahead* Nel progettare sommatori molto veloci talvolta si può ricorrere a strutture ad albero alternative che forniscano un miglior compromesso tra occupazione di area, consumo di potenza e velocità. Di seguito discuteremo brevemente il sommatore di *Brent-Kung* e il sommatore *lookahead di quarto grado*, due delle più comuni architetture alternative.

Materiale protetto da copyright

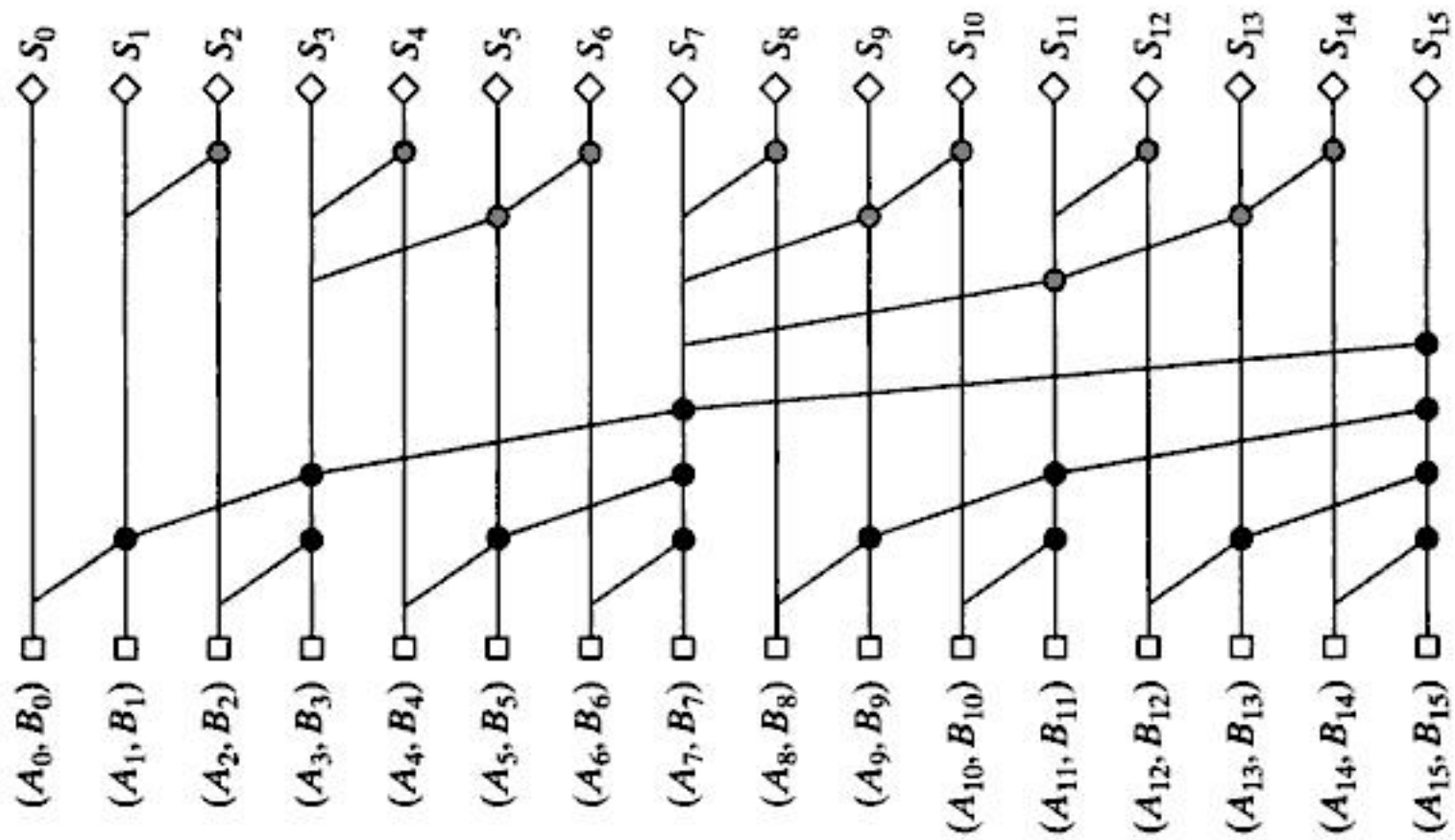


Figura 11-26 Albero di Brent–Kung per un sommatore a 16 bit. L'albero inverso è disegnato in grigio.

L'albero di Kogge-Stone nella Figura 11-22 ha alcune proprietà interessanti: in primo luogo, la sua struttura è regolare, rendendo molto semplice la sua implementazione a livello circuitale; in secondo luogo, il fan-out è praticamente costante attraverso tutto l'albero, soprattutto lungo il cammino critico. Per questi motivi, il dimensionamento dei transistor per ottenere le migliori prestazioni è un'operazione abbastanza semplice e immediata. Allo stesso tempo, tuttavia, la ripetizione della stessa struttura per generare tutti i riporti intermedi avviene al prezzo di un notevole consumo di area a di potenza. Talvolta i progettisti preferiscono penalizzare leggermente la velocità per venire incontro ai limiti imposti dall'occupazione di area o al consumo di energia, scegliendo strutture meno complesse. Un esempio di albero con struttura molto meno complessa è quello che calcola i riporti dei bit la cui posizione è una potenza di due [Brent82], come illustrato nella Figura 11-26 per $N = 16$.

L'albero binario diretto realizza i segnali di riporto solo delle posizioni $2^N - 1$:

$$
\begin{aligned}
(C_{o,0}, 0) &= (G_0, P_0) \cdot (C_{i,0}, 0) \\
(C_{o,1}, 0) &= [(G_1, P_1) \cdot (G_0, P_0)] \cdot (C_{i,0}, 0) = (G_{1:0}, P_{1:0}) \cdot (C_{i,0}, 0) \\
(C_{o,3}, 0) &= [(G_{3:2}, P_{3:2}) \cdot (G_{1:0}, P_{1:0})] \cdot (C_{i,0}, 0) = (G_{3:0}, P_{3:0}) \cdot (C_{i,0}, 0) \\
(C_{o,7}, 0) &= [(G_{7:4}, P_{7:4}) \cdot (G_{3:0}, P_{3:0})] \cdot (C_{i,0}, 0) = (G_{7:0}, P_{7:0}) \cdot (C_{i,0}, 0) \\
&\ldots
\end{aligned}
\tag{11.20}
$$

L'albero binario diretto non è sufficiente a calcolare l'insieme completo dei riporti. Per il calcolo dei riporti rimanenti è necessario l'*albero binario inverso* (disegnato in grigio nella Figura 11-26). Questa struttura combina insieme i risultati intermedi per produrre i riporti rimanenti. Viene lasciata al lettore la dimostrazione che questa struttura esegue correttamente il calcolo dell'intero vettore di riporti. Il circuito sommatore che deriva da questa struttura è

Materiale protetto da copyright

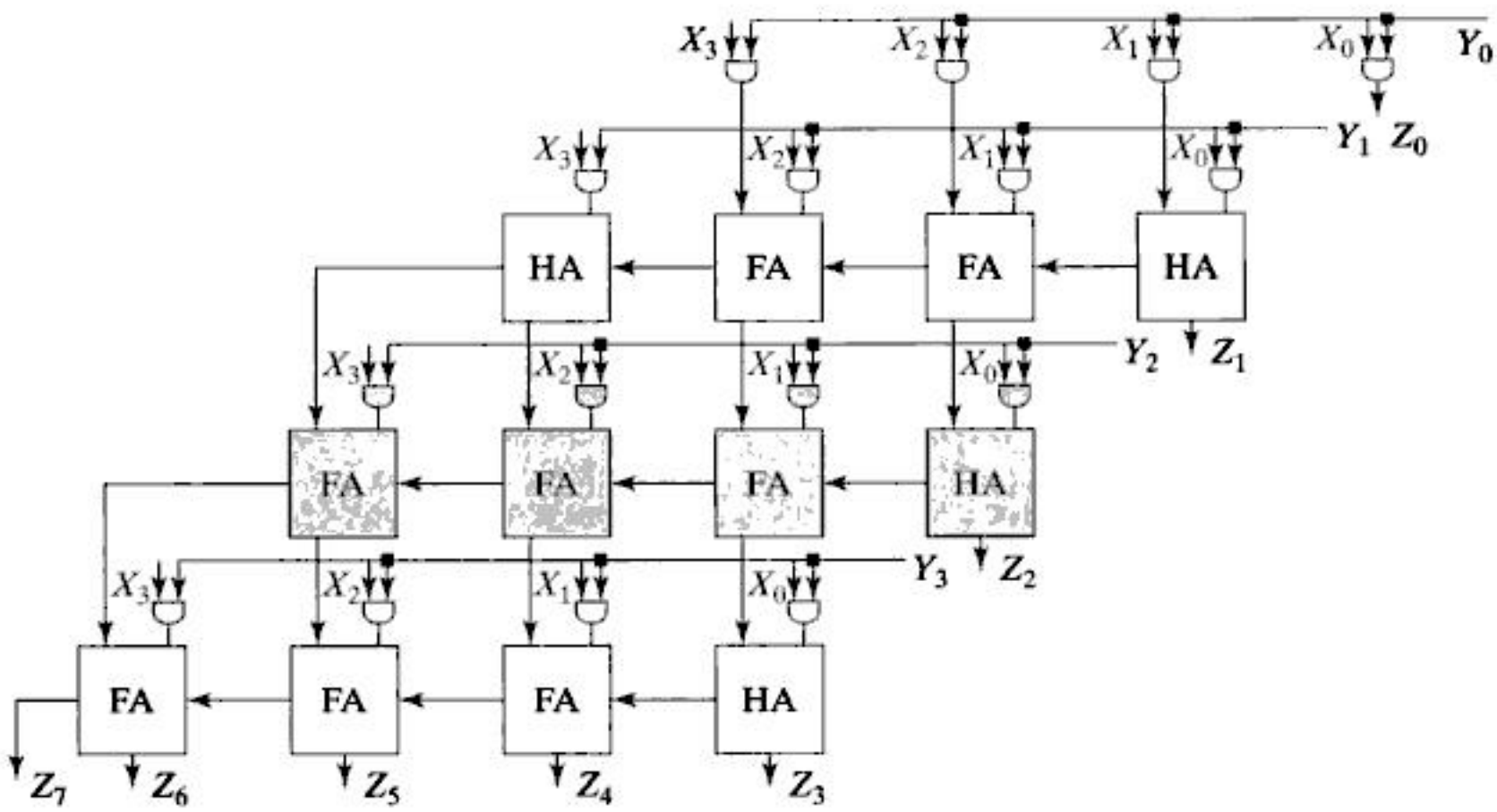


Figura 11-30 Moltiplicatore a matrice per due numeri interi a 4 × 4 bit. HA identifica gli half-adder, cioè un circuito full-adder con solo due ingressi (senza l'ingresso del riporto). La sezione del circuito che realizza una singola somma tra i prodotti parziali è contrassegnata in grigio.

11.4.3 Accumulazione dei prodotti parziali

Dopo che i prodotti parziali sono stati generati essi devono essere sommati. L'accumulazione di queste somme non è nient'altro che un'addizione con più di due addendi. Il modo più semplice per realizzare l'operazione è usare un certo numero di sommatori connessi in modo da formare una matrice: di qui il nome *moltiplicatore a matrice (array multiplier)*. Un modo più sofisticato sarebbe quello di realizzare la somma con una struttura ad albero.

Il moltiplicatore a matrice

La struttura di un moltiplicatore a matrice è illustrata nella Figura 11-30, dalla quale è chiaramente visibile la corrispondenza topologica uno a uno tra gli elementi di questo circuito e la struttura dell'operazione manuale mostrata nella Figura 11-28. La generazione degli N prodotti parziali richiede $N \times M$ porte AND a due ingressi (con la struttura della Figura 11-29).[5] La maggior parte dell'area del moltiplicatore è dedicata all'implementazione degli N prodotti parziali, che richiede $N - 1$ *sommatori a M bit*. La traslazione dei prodotti parziali per ottenere un corretto allineamento è realizzata eseguendo l'opportuna interconnessione delle piste e non richiede l'impiego di ulteriore logica. La struttura fisica del circuito può essere facilmente resa di forma rettangolare, dando luogo a un layout molto compatto ed efficiente.

A causa della struttura a matrice di questo circuito, determinarne il tempo di propagazione è un problema non banale. Consideriamo il moltiplicatore della Figura 11-30, in cui il

[5]Questa particolare implementazione non impiega ovviamente la codifica di Booth. Aggiungendo tale codifica non cambia sostanzialmente l'implementazione. Il numero di sommatori nella matrice sarebbe la metà e la struttura di generazione dei prodotti parziali sarebbe leggermente più complessa.

Materiale protetto da copyright

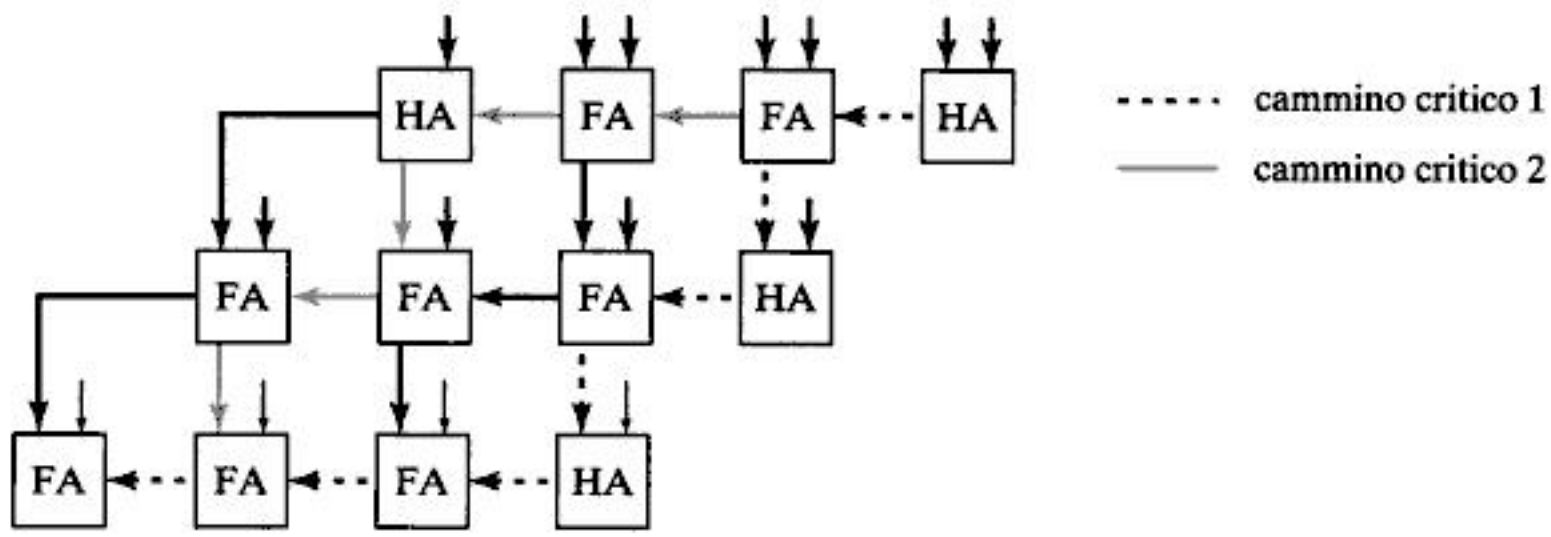


Figura 11-31 Struttura semplificata di un moltiplicatore a matrice con 4 × 4 bit basato sulla propagazione del riporto. Nel disegno sono evidenziati due possibili cammini critici.

circuito di accumulazione delle somme è ottenuto con una serie di sommatori *ripple-carry* (nella figura *HA* sta per half-adder, cioè un circuito full-adder senza il segnale di ingresso del riporto). Per il miglioramento delle prestazioni è necessario innanzitutto identificare il cammino critico. Questo è un problema non banale, poiché esistono molti possibili cammini per il segnale: due di questi sono stati evidenziati nello schema della Figura 11-31. Uno sguardo più da vicino al cammino critico fornisce un'espressione approssimata per il tempo di propagazione (calcolata di seguito per il cammino critico 2). Possiamo esprimere il tempo di ritardo come:

$$t_{molt} \approx [(M-1)+(N-2)]t_C + (N-1)t_S + t_{and} \tag{11.24}$$

dove t_C è il tempo di propagazione tra il riporto di ingresso e il riporto di uscita di ciascun full-adder, t_S è il tempo di propagazione tra il riporto di ingresso e l'uscita della somma del full-adder e t_{and} è il tempo di ritardo della porta AND.

Poiché ciascun cammino ha circa la stessa lunghezza, accelerare uno di essi (ad esempio, realizzando uno dei sommatori con un'architettura più veloce come quella *carry-select*) non avrebbe senso da un punto di vista circuitale. Tutti i cammini critici devono essere trattati nello stesso modo. Dall'Eq. (11.24) possiamo dedurre che per ottimizzare t_{molt} è necessario ridurre sia t_C sia t_S: è quindi auspicabile realizzare il full-adder con t_C uguale a t_S. Questa richiesta è nettamente in contrasto con le caratteristiche delle architetture discusse in precedenza, dove il valore minimo di t_C era la specifica primaria. Un esempio di circuito full-adder con valori paragonabili di t_S e t_C è stato discusso nella Figura 11-7.

Problema 11.8 Moltiplicatore binario con segno

Il moltiplicatore presentato nella Figura 11-30 tratta solo numeri interi senza segno. Modificare la struttura in modo che sia possibile calcolare il prodotto di due numeri interi con segno.

Materiale protetto da copyright

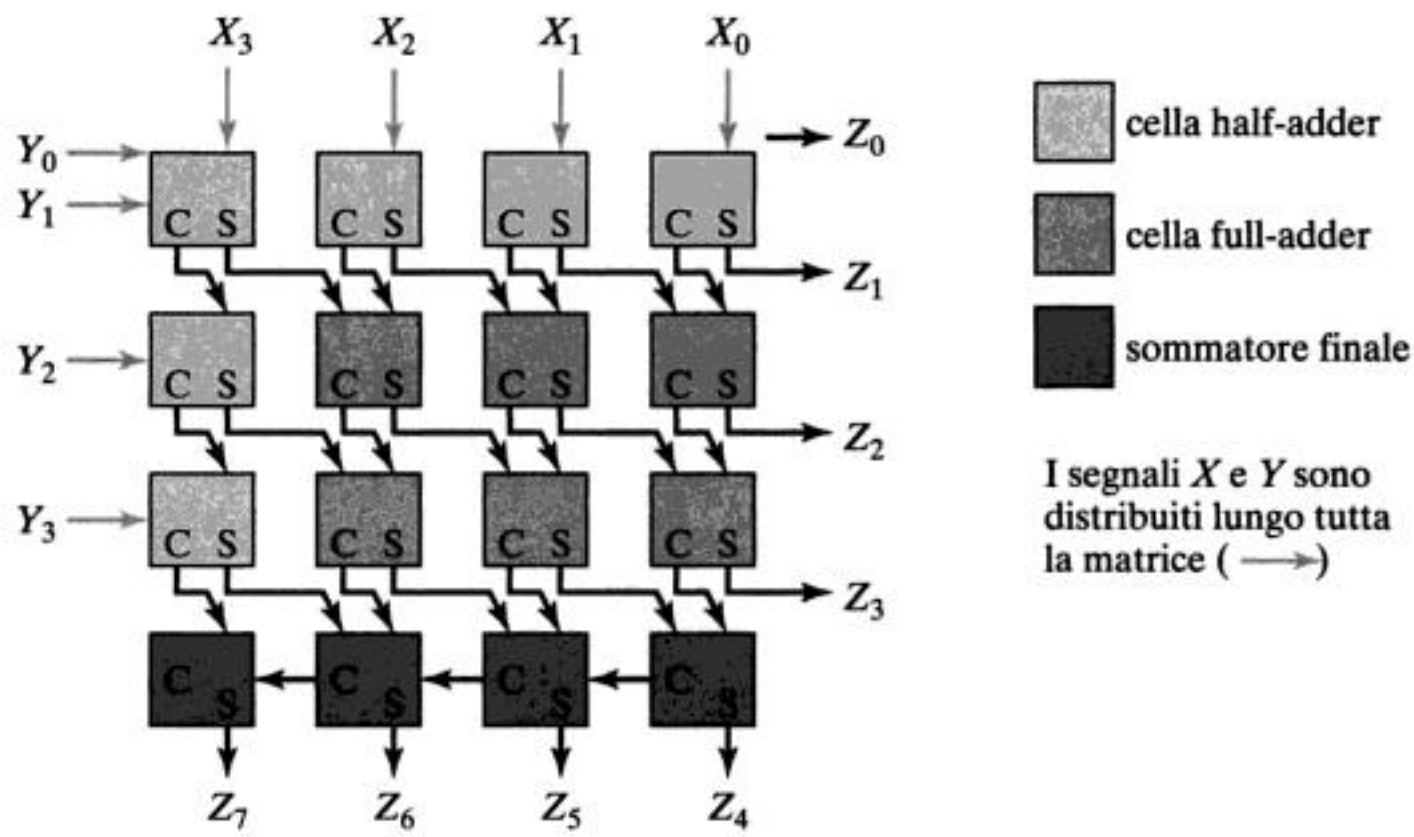


Figura 11-33 Schema che illustra la disposizione dei vari blocchi in un moltiplicatore *carry-save*. I diversi livelli di grigio identificano i diversi tipi di celle (half-adder, full-adder e sommatore finale). L'operazione logica AND tra i segnali X e Y è eseguita prima che i bit arrivino agli ingressi del blocco sommatore.

modulo al resto del chip, è consigliabile che il suo contorno sia approssimativamente rettangolare. La Figura 11-33 mostra schematicamente la disposizione ottimale dei vari blocchi full-adder della matrice che permette di soddisfare questa specifica. Osserviamo la regolarità della topologia che la rende adatta anche alla progettazione assistita da calcolatore.

Il moltiplicatore ad albero

Le somme parziali possono essere anche riorganizzate a forma di albero, riducendo sia il cammino critico sia il numero di full-adder (e half-adder) necessari. Consideriamo il semplice esempio dell'accumulazione dei quattro prodotti parziali a quattro bit ciascuno, come nella Figura 11-34a. Il numero di full-adder necessari all'operazione possono essere ridotti osservando che solo la colonna 3 della matrice somma effettivamente 4 bit, mentre tutte le altre colonne sono meno complesse. Ciò è mostrato nella Figura 11-34b, dove i bit da sommare sono riorganizzati a forma di albero per dimostrare visivamente la profondità variabile dell'accumulazione. La vera sfida è quella di realizzare la matrice con la minima profondità e impiegando il minor numero di elementi sommatori. Il primo tipo di operatore che può essere usato per coprire l'intera matrice è il *full-adder*, che ha tre ingressi (nella stessa colonna) e due uscite: la *somma* collocata nella stessa colonna e il *riporto* collocato nella colonna successiva (a sinistra). Per questo motivo il FA è anche detto un *riduttore 3-2* ed è indicato dal cerchio comprendente tre bit. Il secondo operatore è un *half-adder*, che ha due ingressi (nella stessa colonna) e due uscite (come il *full-adder*). Un HA è indicato da un cerchio che racchiude due bit.

Materiale protetto da copyright

plexer per combinare le operazioni e un bus per riportare all'ingesso il risultato dell'uscita. La prima coppia di multiplexer 9:1 seleziona il dato di ingresso tra 9 sorgenti diverse: tre provenienti dai registri e dalla memoria cache, gli altri sei provenienti direttamente dalle uscite delle sei ALU (il microprocessore di cui parliamo è in grado di eseguire sei istruzioni di elaborazione su numeri interi in un unico ciclo). Uno degli ingressi del multiplexer coincide proprio con il cammino di retroazione che connette l'uscita della ALU ai suoi stessi ingressi, che consente di eseguire in un unico ciclo un'operazione sul risultato precedente. Il secondo multiplexer 5:1 all'ingresso A esegue la traslazione parziale dell'operando, mentre il multiplexer 2:1 all'ingresso B inverte semplicemente l'operando in modo da eseguire la sottrazione. Il sommatore esegue l'addizione $(A + B)$ e la sottrazione $(A - B)$ in complemento a due

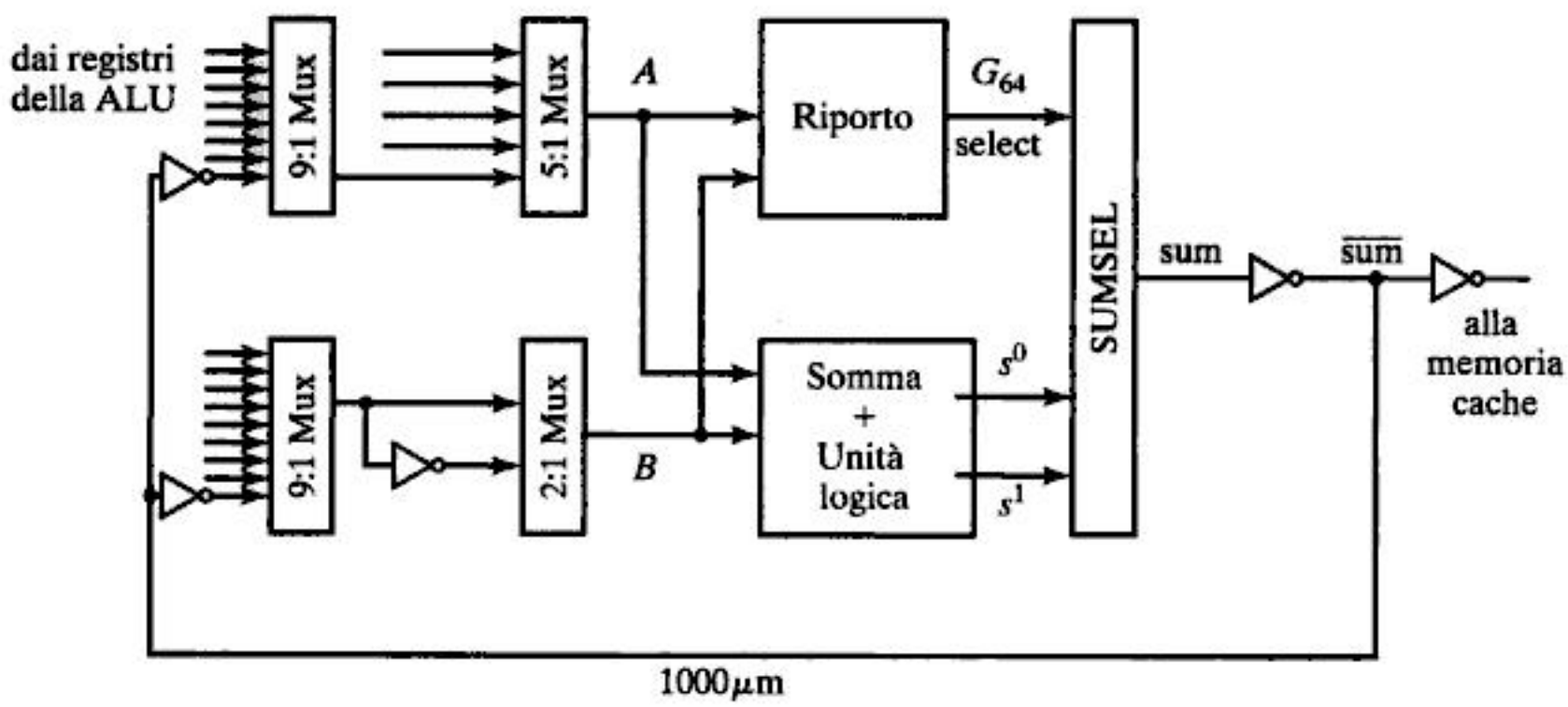


Figura 11-40 Unità logico-aritmetica a 64 bit.

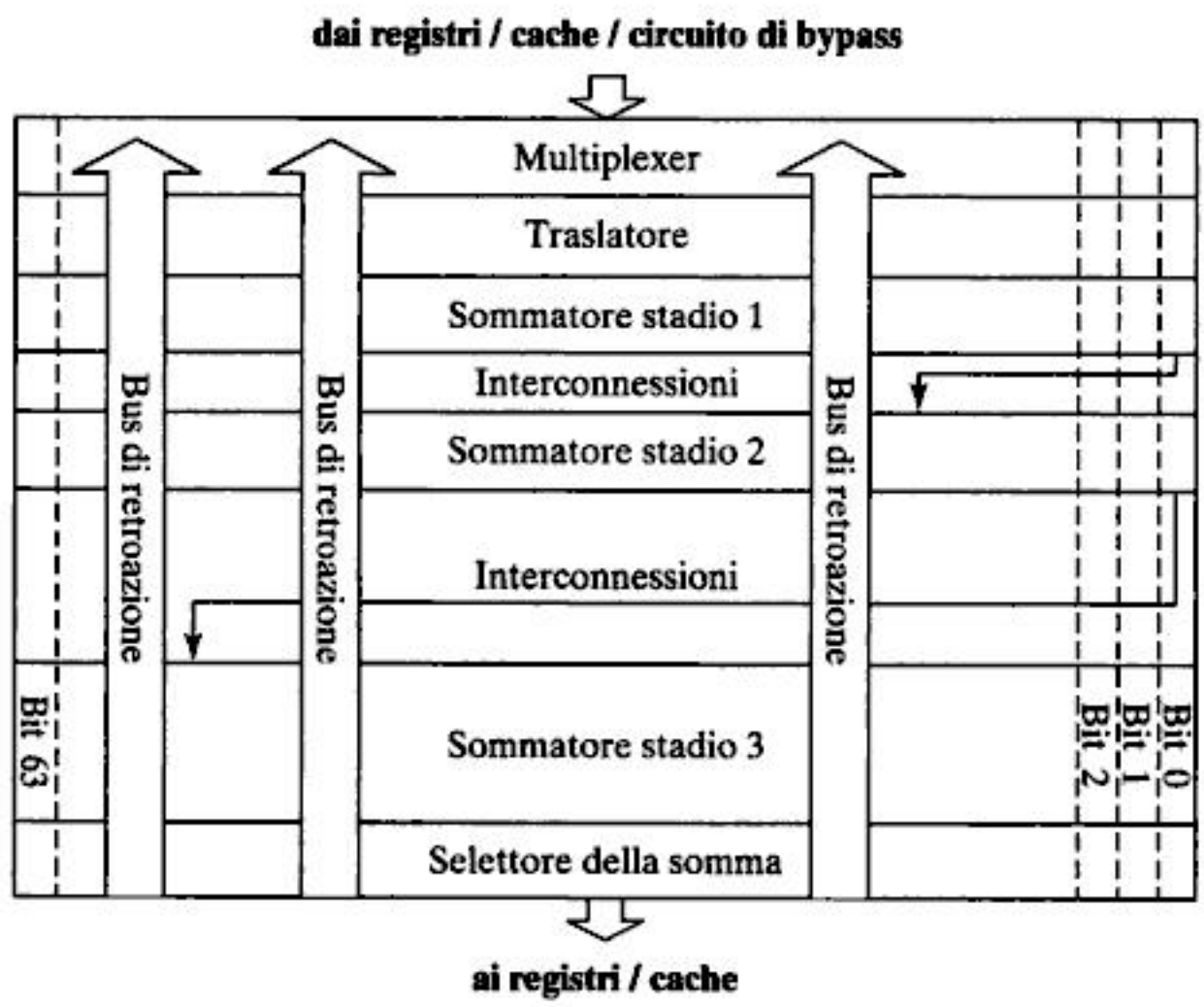


Figura 11-41 Pianta di una ALU a 64 bit.

Materiale protetto da copyright

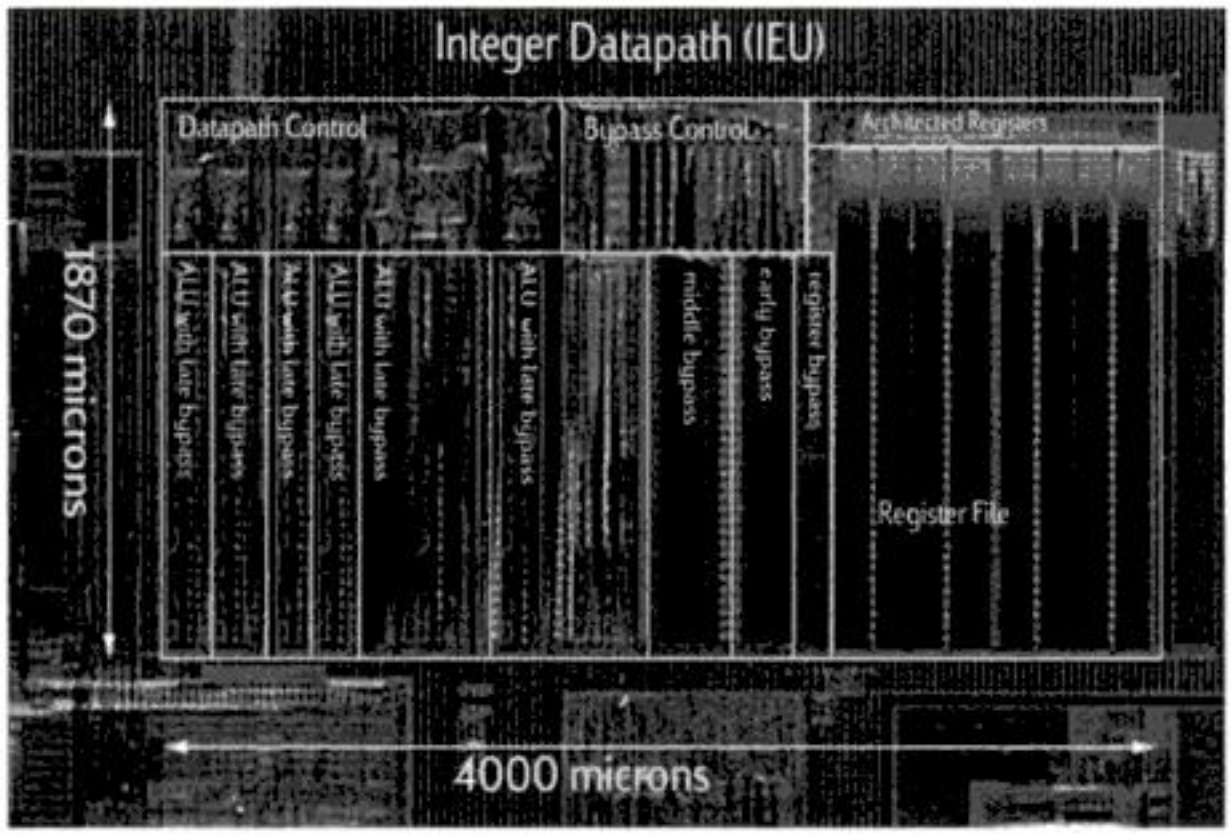


Figura 11-42 Fotografia del processore Itanium, che evidenza la struttura *bit-sliced* delle sei ALU, dei registri e dei circuiti logici di controllo (gentile concessione di Intel).

sugli operandi A e B. In particolare la sottrazione è eseguita negando tutti i bit dell'operando B (usando il multiplexer 2:1) e impostando il riporto di ingresso a 1. Il blocco selettore (SUMSEL) non solo realizza la selezione della somma, ma serve anche a unire le uscite delle unità aritmetiche e logiche. Infine, all'uscita è inserito un buffer per realizzare la retroazione verso il multiplexer di ingresso senza caricare eccessivamente l'uscita della ALU. Ad esempio, per un processore come quello Intel/HP Itanium [Fetzer02], il bus dati deve interconnettere tutte le sei unità, con piste lunghe fino a 2 mm in una tecnologia da 0.18 μm. Se poi aggiungiamo il carico proveniente dai vari registri dati, ciascuna linea del bus presenta un carico capacitivo di circa 0.5 pF.

Una ALU rappresenta un esempio tipico della struttura *bit-sliced*. Ciascuna linea di bit in ciascun blocco è fatta combaciare con la stessa linea dei blocchi vicini, in modo tale da minimizzare la lunghezza delle interconnessioni verticali. La pianta di questa ALU a 64 bit è mostrata nella Figura 11-41. Le interconnessioni tra le diverse celle (relative a bit diversi) è fatta orizzontalmente in metal-3, ad eccezione delle lunghe piste orizzontali tra gli stadi sommatori 1 e 2 e tra gli stadi sommatori 2 e 3, che sono disegnate combinando tratti in metal-3 e metal-4. Ad esempio, se il sommatore è implementato con architettura *carry-lookahead del quarto ordine*, la pista di interconnessione più lunga dopo il secondo stadio di segnali *propagate* e *generate* attraversa 48 celle.

La pista di interconnessione che realizza l'anello di retroazione uscita-ingresso è realizzata sul livello di interconnessione più alto (ad esempio metal-5). Nel caso in cui il microprocessore abbia più di una ALU, questa linea si estende lungo tutte le ALU, come è possibile osservare nella Figura 11-42. Il numero di anelli di retroazione determina spesso la spaziatura tra i bit nei processori con molteplici ALU. Nel nostro esempio, nove bus di 64 bit ciascuno devono incrociare la ALU. Per non dover introdurre ulteriori piste, il layout è fatto in modo da far coincidere la larghezza di una linea di bit con la spaziatura tra le nove piste del bus. Essendo molto lunghe, queste piste sono disegnate con larghezza e spaziatura almeno doppia per ridurre la le resistenze e le capacità parassite. ■

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

- **Sede di implementazione.** Molte tecniche sono implementate direttamente in *sede di progetto*. La larghezza e la lunghezza di canale dei transistor, per esempio, sono stabilite nell'istante in cui il transistor stesso è disegnato. Al contrario, la tensione di alimentazione e la tensione di soglia possono essere sia fissate staticamente in fase di progetto, sia cambiate dinamicamente *durante il funzionamento* del sistema. Altre tecniche gestiscono principalmente gli intervalli di tempo durante i quali un modulo del sistema è in fase di inattività o attesa (*standby*), mirando a ridurre al minimo il consumo statico di tali moduli quando non sono operativi.
- **Contributo di consumo verso cui sono orientate.** Un'altro tipo di classificazione delle tecniche di gestione del consumo riguarda il contributo di consumo che esse mirano a ridurre: *consumo attivo* (dinamico) o le *correnti di perdita* (consumo statico). Abbassare la tensione di alimentazione, ad esempio, è una tecnica molto allettante: non solo riduce l'energia dissipata ad ogni transizione in modo quadratico (sebbene a discapito delle prestazioni) ma diminuisce anche le correnti di perdita; d'altra parte, aumentare la tensione di soglia dei transistor influenza soprattutto il consumo statico.

Il tempo di inattività di un blocco del sistema merita un'attenzione particolare. Se un circuito digitale continua a ricevere il segnale di clock, mentre è in fase di attesa, la sua rete di distribuzione del clock e i relativi registri continuano a consumare energia, anche se non viene eseguita alcuna operazione. Si ricordi che circa un terzo del consumo di energia in un sistema digitale avviene a causa della rete di distribuzione del clock. Un modo abbastanza comune per ridurre il consumo di potenza durante la fase di inattività è la tecnica *clock gating* introdotta nel Capitolo 10. In tale approccio, la connessione principale tra il blocco in questione e il segnale di clock viene disabilitata (aperta) quando il blocco non è operativo. Tuttavia, questa tecnica non ha alcun effetto sul consumo statico dovuto alle correnti di perdita. Per agire anche sul consumo statico è necessario utilizzare architetture ancor più complicate che permettano di aumentare la tensione di soglia o isolare il blocco dalla rete di alimentazione.

Nei prossimi paragrafi discuteremo in dettaglio tutte queste tecniche applicabili sia staticamente in fase di progetto sia dinamicamente durante il funzionamento del sistema.

11.7.1 Tecniche di riduzione del consumo in fase di progetto

Riduzione della tensione di alimentazione

Ridurre la tensione di alimentazione comporta una diminuzione quadratica del consumo di potenza dinamico ed è quindi la tecnica più allettante. D'altro canto, il tempo di ritardo delle porte logiche CMOS è inversamente proporzionale alla tensione di alimentazione. A livello di struttura questa perdita in prestazioni può essere compensata per altre vie, come l'ottimizzazione logica e dell'architettura del sistema. Ad esempio, un sommatore *ripple-carry* potrebbe essere sostituito con una struttura più veloce, come un sommatore *lookahead*. Questa seconda architettura è molto più complessa, occupa più area e ha una maggior capacità fisica e logica. Comunque, questi fattori sono più che compensati dal fatto che il sommatore

Materiale protetto da copyright

può lavorare con una tensione di alimentazione sostanzialmente minore, ottenendo le stesse prestazioni.

Il compromesso tra i sommatori *ripple-carry* e *lookahead* avviene a livello logico. Similmente, anche l'ottimizzazione architetturale può essere impiegata per compensare la riduzione di V_{DD} in modo ancora più efficiente, come illustra l'Esempio 11.7.

Esempio 11.7 Riduzione del consumo di potenza impiegando il parallelismo

Per mostrare come l'architettura di un sistema può essere usata per consentire la riduzione della tensione di alimentazione, compensando la riduzione della velocità, consideriamo una semplice unità di elaborazione a 8 bit costituita da un sommatore e un comparatore, in tecnologia CMOS da 2 μm [Chandrakasan92]. Come mostrato nella Figura 11-43, gli ingressi A e B sono sommati e il risultato è confrontato con l'ingresso C. Supponiamo che il tempo di ritardo al caso peggiore attraverso il sommatore, il comparatore e i registri sia approssimativamente 25 ns con una tensione di alimentazione di 5 V. Nel migliore dei casi il circuito può essere fatto funzionare con un periodo di clock T = 25 ns. Se le specifiche richieste impongono che il sistema lavori alla massima frequenza, è chiaro che la tensione di alimentazione non può essere ridotta ulteriormente, non essendo tollerabile nessun ritardo aggiuntivo. Usando questa topologia come riferimento, cerchiamo di ottimizzare l'architettura dell'unità di elaborazione per ridurre il consumo. Il consumo dinamico medio di potenza della struttura di riferimento è:

$$P_{ref} = C_{ref}\, V_{ref}^2\, f_{ref} \tag{11.26}$$

dove C_{ref} è il valore effettivo della capacità logica del sistema (prodotto tra capacità fisica e fattore di attività) mediamente caricata e scaricata ad ogni ciclo. Il valore effettivo della capacità può essere determinato mediando l'energia consumata su un grande numero di cicli di clock inviando in ingresso una sequenza casuale di dati con distribuzione statistica uniforme.

Mantenere costante il flusso di informazioni, pur riducendo la tensione di alimentazione è possibile se si impiega un architettura in parallelo, come illustra la Figura 11-44: due unità identiche (sommatore e comparatore) sono connesse in parallelo, permet-

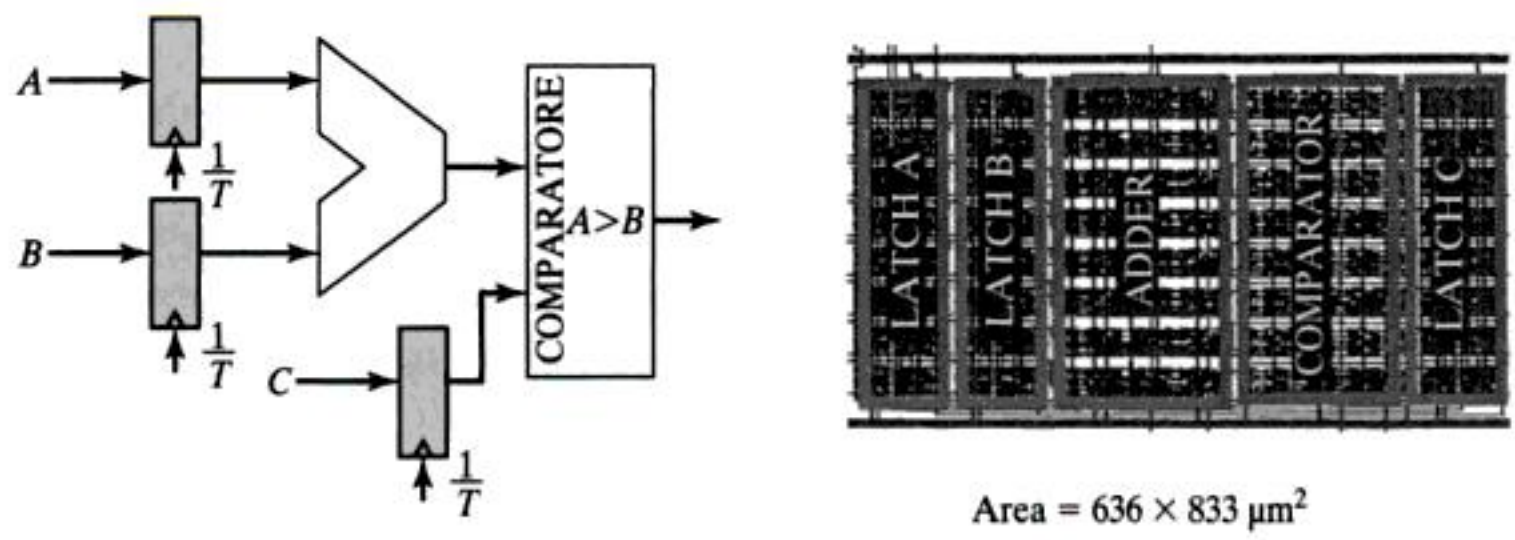


Figura 11-43 Esempio di unità di elaborazione e corrispondente layout.

Materiale protetto da copyright

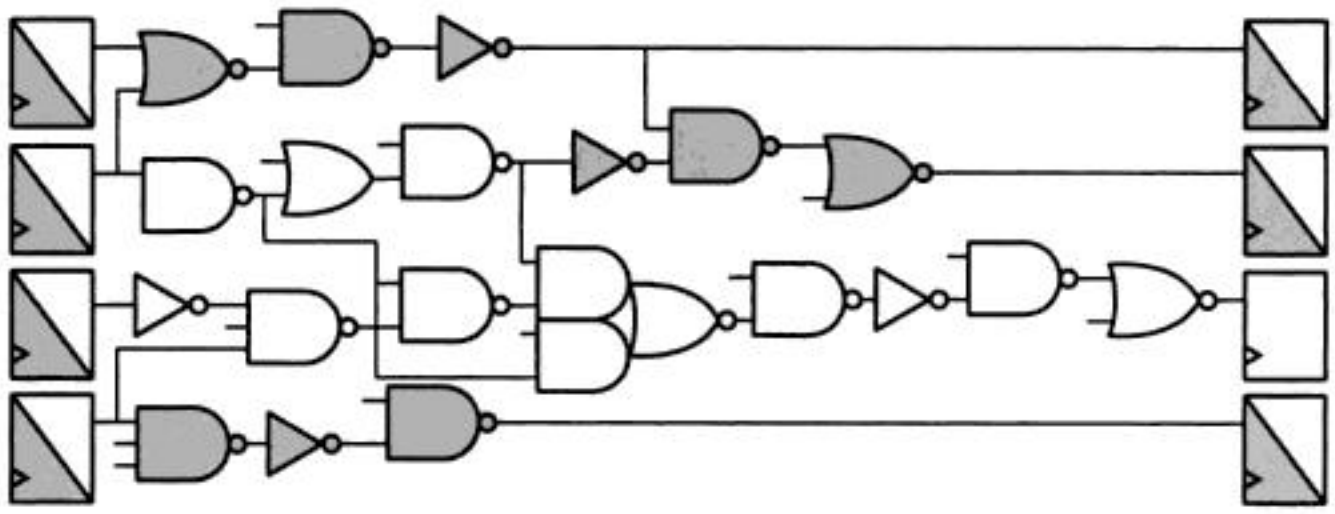


Figura 11-48 Circuito che impiega due tensioni di alimentazione usando la tecnica di *riduzione della tensione a blocchi*. Ciascun percorso inizia con una tensione di alimentazione alta e passa al valore più basso (porte logiche colorate in grigio), se si creano tempi morti. Il circuito di adattamento del livello logico è inserito all'interno dei registri.

due ragioni: in primo luogo, molte porte logiche sono condivise tra due o più cammini; in secondo luogo, non è per niente pratico generare e distribuire un gran numero di tensioni di alimentazioni differenti. Una soluzione più semplice, che impiega solo due tensioni di alimentazione è illustrata nella Figura 11-48. Usando una tecnica di *riduzione della tensione a blocchi*, ciascun cammino inizia con una tensione di alimentazione alta e passa ad una tensione più bassa non appena si creano lungo il cammino dei *tempi morti*. La conversione tra livelli logici è eseguita nei registri alla fine del cammino con un circuito come quello mostrato nella Figura 11-47. Notiamo che la conversione del livello di tensione è necessaria solo se lo stadio logico che segue non è in grado di leggere completamente i livelli logici corrispondenti alla tensione di alimentazione bassa.

L'effetto di questo metodo è illustrato nella Figura 11-49, che mostra la distribuzione dei tempi di ritardo per un tipico blocco con tensione di alimentazione singola e doppia. Usando una singola alimentazione, gran parte dei percorsi dei segnali ha un tempo di propagazione considerevolmente minore del cammino critico. Aggiungendo una seconda fonte di alimentazione, il massimo della distribuzione si avvicina al ritardo più lungo, rendendo critici molti più cammini. Se il circuito originale ha molto pochi cammini critici, l'adozione di questa tecnica porta ad un considerevole risparmio di energia. Al contrario, in un blocco in cui i quasi tutti i cammini sono già critici (o quasi), introdurre una ulteriore sorgente di alimenta-

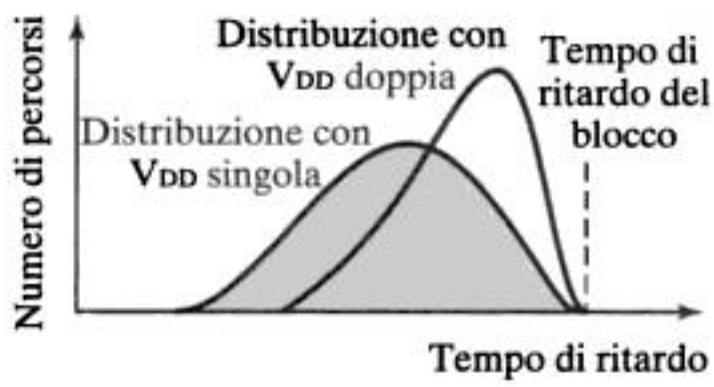


Figura 11-49 Distribuzione tipica dei tempi di ritardo associati ai cammini di un blocco, usando una o due tensioni di alimentazione diverse. Il valore del tempo di ritardo è riportato nell'asse *x*, mentre l'asse y rappresenta il numero di cammini con un dato tempo di propagazione.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

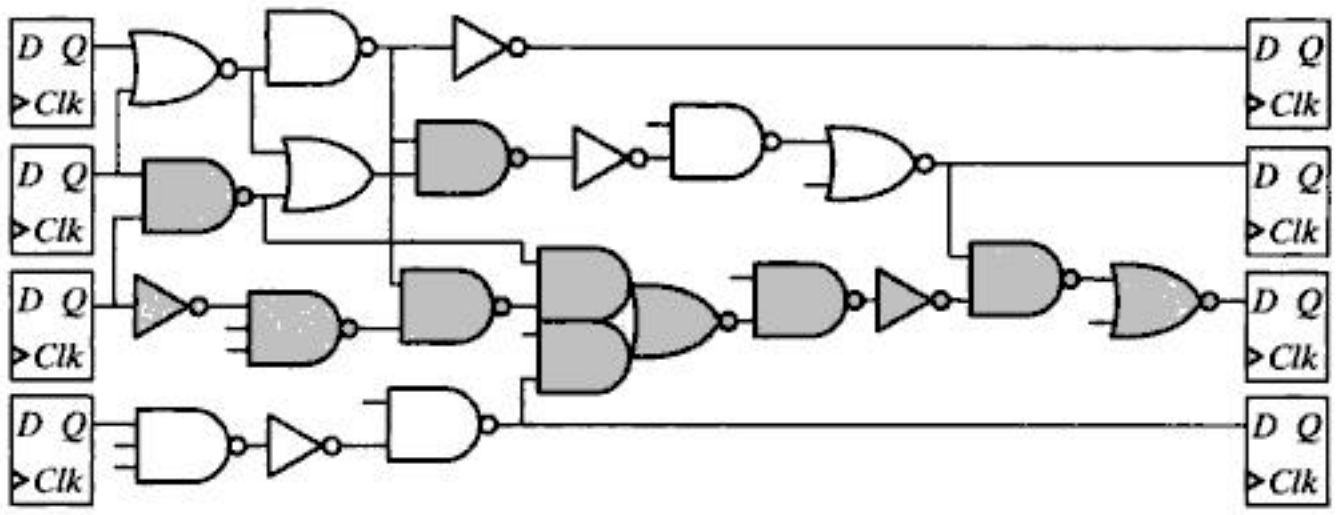


Figura 11-51 Esempio di circuito realizzato con due tipi di transistor caratterizzati da un diverso valore di tensione di soglia. Le porte logiche in grigio sono costruite usando dispositivi con tensione di soglia più bassa e sono impiegati solo lungo il cammino critico.

Impiego di dispositivi con tensioni di soglia differenti

L'impiego di dispositivi con tensione di soglia diversa offre un altro modo per ottenere un compromesso tra velocità e consumo. La maggior parte delle tecnologie CMOS con lunghezza di canale inferiore a 0.25 μm mettono a disposizione due tipi di transistor (sia di tipo *n* si di tipo *p)* con tensioni di soglia che differiscono di circa 100 mV. I dispositivi con tensione di soglia maggiore sono caratterizzati da una corrente di perdita in regione di sottosoglia di un ordine di grandezza più bassa rispetto ai dispositivi con tensione di soglia minore, al prezzo di una riduzione del 30% della corrente in regione attiva. I transistor con tensione di soglia minore sono impiegati preferibilmente lungo i cammini critici, mentre i transistor con tensione di soglia più alta sono usati in tutti gli altri blocchi. La scelta sul tipo di transistor può essere fatta anche cella per cella, anziché transistor per transistor, semplificando il progetto del circuito [Wei99, Kato00]. È importante notare che usare transistor con tensione di soglia differente non richiede alcun adattatore di livello o altri circuiti particolari, come illustrato nella Figura 11-51. Non è neppure necessario il raggruppamento delle celle come è stato fatto per la tensione di alimentazione, poiché il maggior numero possibile di transistor possono essere facilmente convertiti con tensione di soglia elevata finché tutti i tempi morti sono stati completamente eliminati. Un accurata scelta delle tensioni di soglia dei transistor può ridurre il consumo statico di potenza dell'80% – 90%.

Sebbene l'uso di tensioni di soglia multiple abbia come obiettivo principale la riduzione del consumo statico (dovuto alle correnti di perdita), porta anche un modesto vantaggio in termini di consumo dinamico, per due ragioni fondamentali: la riduzione della capacità gate-canale in condizioni di interdizione e la riduzione dell'escursione di tensione dei nodi interni alle porte logiche (V_{DD}–V_{ThH}). Questa riduzione è tuttavia parzialmente compensata dall'aumento dei contributi di bordo delle capacità associate alle giunzioni di drain e di source. La diminuzione netta del consumo dinamico è stimata intorno al 4%.

Ridurre la capacità mediante il dimensionamento dei transistor

La capacità di ingresso di una porta logica CMOS è direttamente proporzionale alla dimensione dei transistor che la compongono e quindi alla sua velocità. Nel Capitolo 6, abbiamo esaminato il dimensionamento ottimale delle porte concludendo che ciascuno stadio

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

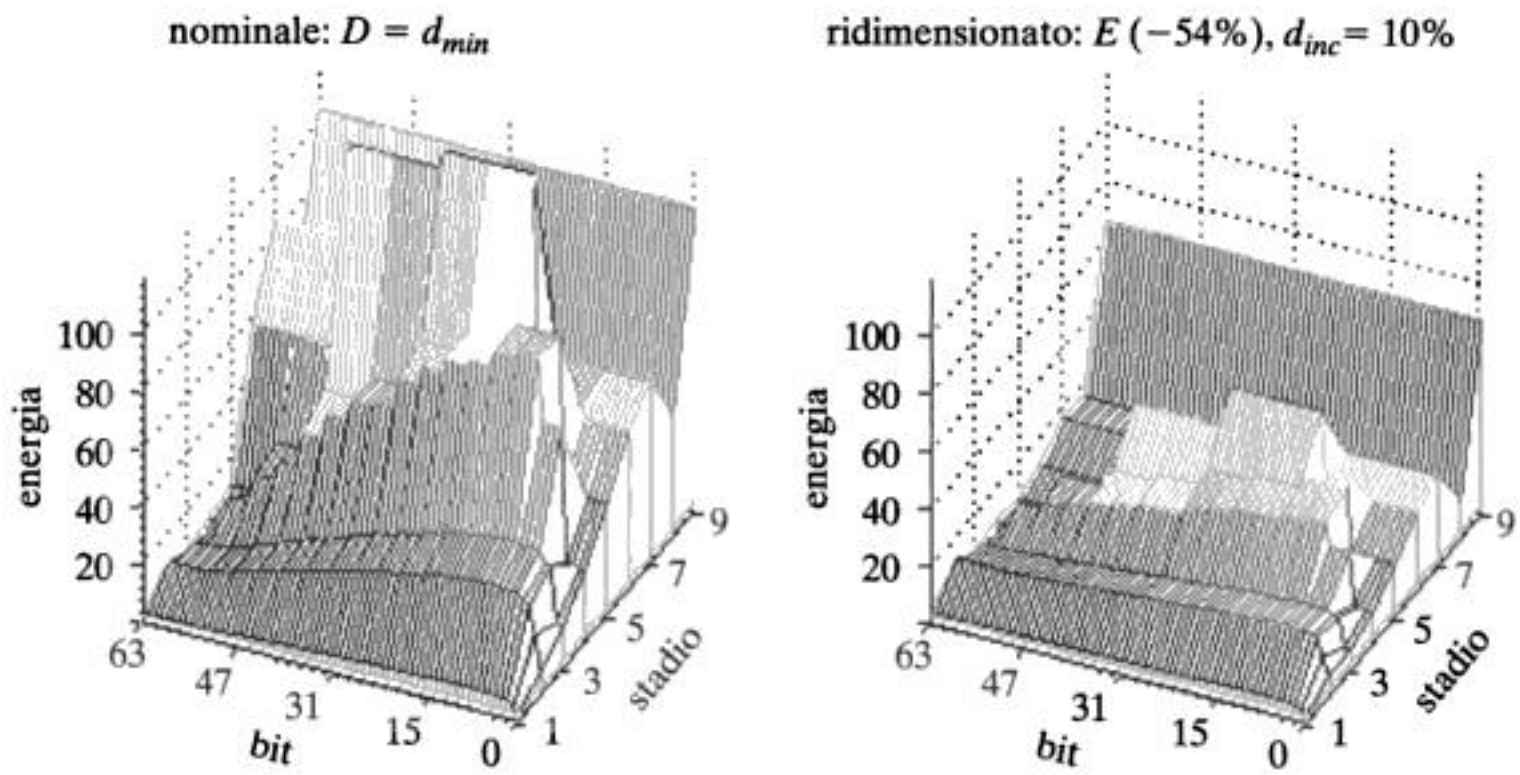


Figura 11-52 Consumo di energia in un sommatore a 64-bit adder: (a) dimensionato in modo da ottenere il minimo ritardo, (b) dimensionato per ottenere un ritardo del 10% superiore al minimo e ottimizzato per ridurre il consumo di potenza.

zionare tutti i cammini del sommatore e renderli uguali al cammino critico (in termini di ritardo). Poiché i cammini attraverso un sommatore corrispondono all'incirca alle bit-slice, collochiamo ciascuna porta logica del sommatore in una bit-slice: ci sono 64 bit-slice e 9 stadi logici. Questa ripartizione è adatta per la rappresentazione della Figura 11-52a che mostra la mappa del consumo di energia in condizioni di minimo ritardo. Osserviamo come il sommatore abbia il consumo massimo lungo il cammino critico. La Figura 11-52b mostra il profilo del consumo di energia dello stesso sommatore, ma questa volta ridimensionato in modo tale che il tempo di ritardo sia il 10% maggiore di quello ottimale. L'energia dissipata nell'intero sommatore è ridotta del 54%. Al contrario l'adozione di una doppia tensione di alimentazione comporterebbe un risparmio notevolmente inferiore (solo del 27%), mentre la semplice riduzione uniforme della tensione di alimentazione ridurrebbe il consumo del 22% [Stojanovic02]. In sintesi, **ridimensionare i transistor è un metodo molto efficiente per la riduzione del consumo di potenza dentro i blocchi logici, dove la maggior parte della dissipazione di energia avviene all'interno del blocco piuttosto che per caricare e scaricare la capacità di carico esterna.**

Ridurre il fattore di attività mediate l'ottimizzazione logica e strutturale

Dato che la capacità effettiva deriva dal prodotto tra la capacità fisica e il fattore di attività, è raccomandabile ridurre entrambi questi fattori. Esistono svariati metodi per ridurre il fattore di attività di un sistema senza che siano compromesse le prestazioni. Alcune di queste sono state già discusse nel Capitolo 6. Per brevità discuteremo solo un piccolo insieme dei metodi più rappresentativi per l'ottimizzazione logica del circuito (consultare [Rabaey96] e [Chandrakasan98] per una panoramica più approfondita).

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

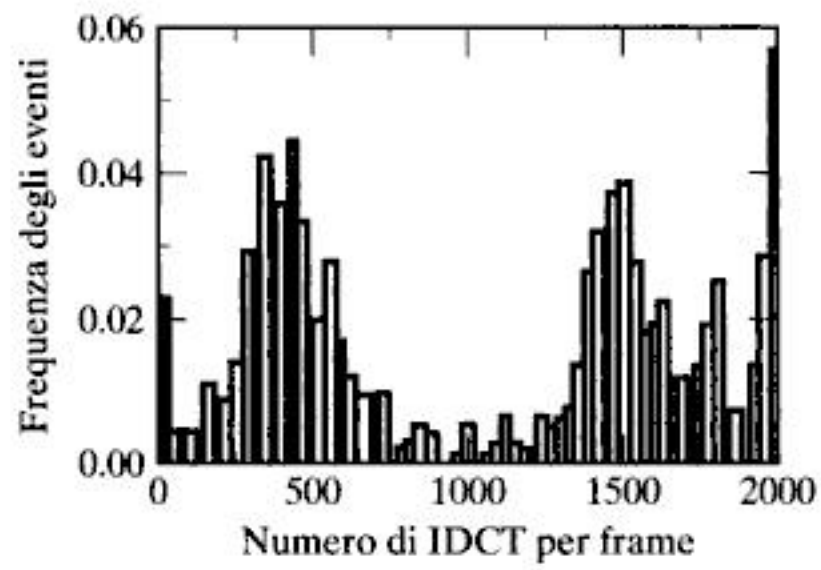


Figura 11-55 Istogramma del numero occorrenze dell'algoritmo IDCT per un tipico decodificatore MPEG.

non sono richieste continuamente le massime prestazioni. Consideriamo, ad esempio, un microprocessore ad uso generale impiegato in applicazioni portatili, come i notebook, le agende elettroniche e i telefoni cellulari. Le funzioni computazionali che devono essere eseguite su un tale processore cadono all'interno di tre categorie principali: elaborazioni con calcoli intensivi, funzioni a bassa velocità, e operazioni in stato di attesa. Le elaborazioni che richiedono calcoli intensivi e, in generale, tutte le operazioni con un vincolo in *latenza*, hanno bisogno di tutta la capacità computazionale del processore per eseguire le operazioni in tempo reale. Esempi di ciò sono la decompressione audio e video MPEG. Le operazioni caratterizzate da flussi lenti e lunga latenza, come la compressione del testo, l'inserimento dei dati e le operazioni di backup in memoria, hanno tempi di completamento molto rilassati e richiedono solo una frazione delle massime prestazioni di un microprocessore. Non essendoci alcun vantaggio nel concludere l'operazione velocemente, finire l'elaborazione con troppo anticipo può essere considerato come spreco di energia. Infine le applicazioni portatili passano la maggior parte del tempo in stato di attesa, aspettando che un'azione dell'utente o un evento esterno. In sintesi le esigenze di flusso e latenza di un'applicazione portatile varia drasticamente nel corso del tempo.

Le stesse operazioni che richiedono una massiccia mole di calcoli, come la codifica MPEG, richiedono prestazioni variabili nel tempo mentre elaborano sequenze tipiche di dati. Ad esempio, il numero di volte che un decodificatore MPEG calcola la coseno-trasformata inversa discreta (*IDCT, inverse discrete cosine-transform*) per ciascun frame video varia moltissimo in funzione della quantità di movimenti del filmato. Ciò è illustrato nella Figura 11-55, che mostra la distribuzione del numero di *IDCT per frame* per una tipica sequenza video. Il processore che deve eseguire questo algoritmo è soggetto ad un carico di lavoro variabile da frame a frame.

Abbassare la frequenza di clock mentre si eseguono i carichi di lavoro più leggeri, riduce il consumo di potenza, ma non risparmia energia: tutte le operazioni sono ancora eseguite con la stessa tensione di alimentazione. Tuttavia, se viene abbassata sia la frequenza di clock sia la tensione di alimentazione, il consumo di energia si riduce. Allo scopo di mantenere alto il flusso di dati, quando il carico di lavoro è elevato, ma ridurre al minimo il consumo di energia, quando il carico di lavoro è basso, è necessario poter variare dinamicamente

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

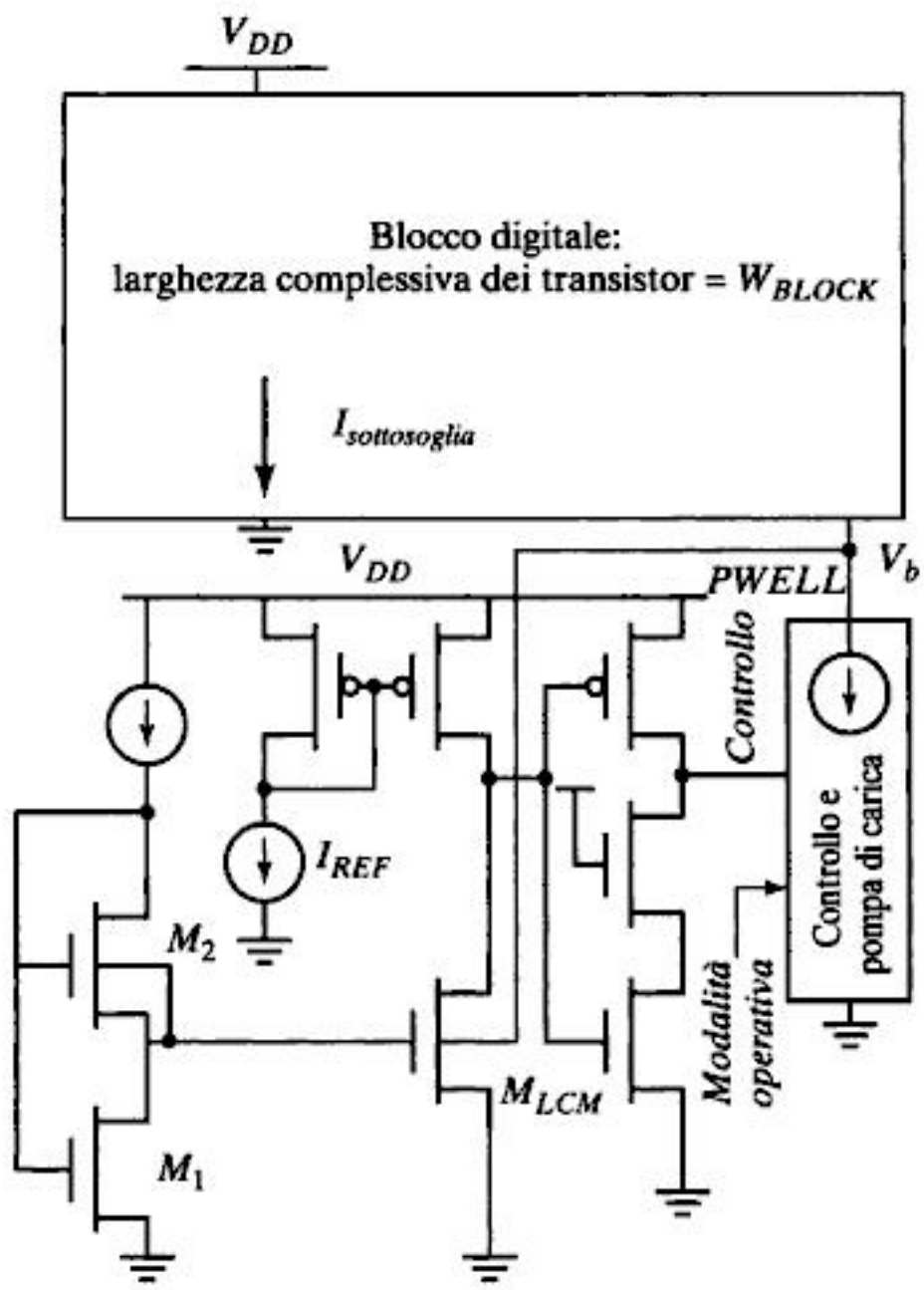


Figura 11-61 Esempio di circuito che implementa il controllo dinamico della tensione di soglia mediante la misura della corrente di perdita [Kuroda96].

mente differenziare la polarizzazione dei substrati dei transistor cella per cella richiede costi e complessità di layout maggiori.

Analogamente alla gestione dinamica della tensione di alimentazione, anche la *gestione dinamica della tensione di soglia (DTS, Dynamic Threshold Scaling)* è un sistema basato su un anello di retroazione, che può essere usato per vari scopi:

- ridurre le correnti di perdita in fase di inattività o di attesa;
- compensare le variazioni di tensione di soglia lungo il chip durante il normale funzionamento del circuito;
- rallentare il flusso di dati del circuito per ridurre sia il consumo dinamico sia il consumo statico in funzione delle prestazioni desiderate.

Poiché il flusso di corrente attraverso il substrato è di gran lunga inferiore a quello nelle linee di alimentazione, il sistema DTS ha un'occupazione di area molto minore del DVS. Un sistema a retroazione, progettato per il controllo della corrente in un modulo digitale, è illustrato nella Figura 11-61 [Kuroda96 e Kuroda01] e consiste di un circuito di controllo della corrente di perdita e un circuito di polarizzazione del substrato realizzato mediante una pompa di carica, inserito nel modulo logico di interesse.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

[Oklobdzija01] V. G. Oklobdzija, "High-Speed VLSI Arithmetic Units: Adders and Multipliers," in *Design of High-Performance Microprocessor Circuits*, Chandrakasan, Bowhil, Fox (eds.), IEEE Press, 2001.

[Omondi94] A. Omondi, *Computer Arithmetic Systems*, Prentice Hall, 1994.

[Pering00] T. A. Pering, *Energy-Efficient Operating System Techniques*, Ph.D. Dissertation, University of California, Berkeley, 2000.

[Rabaey96] J. Rabaey, and M. Pedram, Ed., "Low-Power Design Methodologies," Kluwer Academic Publishers, 1996.

[Stojanovic02] V. Stojanovic, D. Markovic, B. Nikolic, M. A. Horowitz, and R. W. Brodersen, "Energy-Delay Tradeoffs in Combinational Logic using Gate Sizing and Supply Voltage Optimization," *European Solid-State Circuits Conference, ESSCIRC'2002*, Florence, Italy, September 24–26, 2002.

[Suzuoki99] M. Suzuoki *et al.*, "A Microprocessor with a 128-bit CPU, Ten Floating-Point MAC's, Four Floating-Point Dividers, and an MPEG-2 Decoder," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 11, pp. 1608–1618, November 1999.

[Swartzlander90] E. Swartzlander, ed., *Computer Arithmetic—Part I and II*, IEEE Computer Society Press, 1990.

[Usami95] K. Usami, and M. Horowitz, "Clustered Voltage Scaling Technique for Low-Power Design," *Proceedings, 1995 International Symposium on Low Power Design*, Dana Point, CA, pp. 3–8, April 1995.

[Vermassen86] H. Vermassen, "Mathematical Models for the Complexity of VLSI," (in dutch), Master's Thesis, Katholieke Universiteit, Leuven, Belgium, 1986.

[Wallace64] C. Wallace, "A Suggestion for a Fast Multiplier," *IEEE Trans. on Electronic Computers*, EC-13, pp. 14–17, 1964.

[Weinberger56] A. Weinberger and J. L. Smith, "A One-Microsecond Adder Using One-Megacycle Circuitry," *IRE Transactions on Electronic Computers*, vol. 5, pp. 65–73, 1956.

[Wei99] L. Wei, Z. Chen, K. Roy, M.C. Johnson, Y. Ye, and V. K. De, "Design and Optimization of Dual-Threshold Circuits for Low-Power Applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 7, no. 1, pp. 16–24, March 1999.

[Weinberger81] A. Weinberger, "A 4:2 Carry-Save Adder Module," *IBM Technical Disclosure Bulletin*, vol. 23, January 1981.

[Weste85&93] N. Weste and K. Eshragian, *Principles of CMOS VLSI Design: A Systems Perspective*, 2d ed., Addison-Wesley, 1985 and 1993.

Esercizi

Per una collezione aggiornata di esercizi e problemi relativi ai blocchi aritmetici consultare la pagina web **http://bwrc.eecs.berkeley.edu/IcBook.**

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

tutte le parole nella memoria CAM caratterizzate dalla sequenza 0×123 nei bit più significativi, dobbiamo scrivere le parole 0×12300000 nel *comparando* e $0 \times$ FFF00000 nella *maschera*. Tutte le 512 righe della CAM sono simultaneamente abilitate, confrontando il dato in esse contenuto con il *comparando*, tenendo conto dei 12 bit più significativi (come indicato nella *maschera*). Tutte le righe che soddisfano il confronto sono passate al blocco di validazione. Poiché le righe che contengono dati non validi non sono rilevanti (ciò accade generalmente quando la memoria non è completamente piena), solo le righe valide che contengono la sequenza di bit desiderata vengono passate al blocco codificatore (*priority encoder*). Nell'eventualità in cui il *comparando* sia trovato in due o più righe, il loro indirizzo nella matrice della memoria CAM costituisce il parametro discriminante. Per fare ciò, il codificatore considera tutte le locazioni di memoria che contengono la parola cercata, sceglie quella con indirizzo più alto, e la codifica in formato binario.[2] Essendovi 512 righe nella memoria CAM, sono indispensabili 9 bit per definirne l'indirizzo. Inoltre è necessario un bit aggiuntivo per indicare se è la sequenza è stata trovata, poiché potrebbe verificarsi il caso in cui nessuno dei dati nella CAM soddisfi la sequenza di bit nel *comparando*.

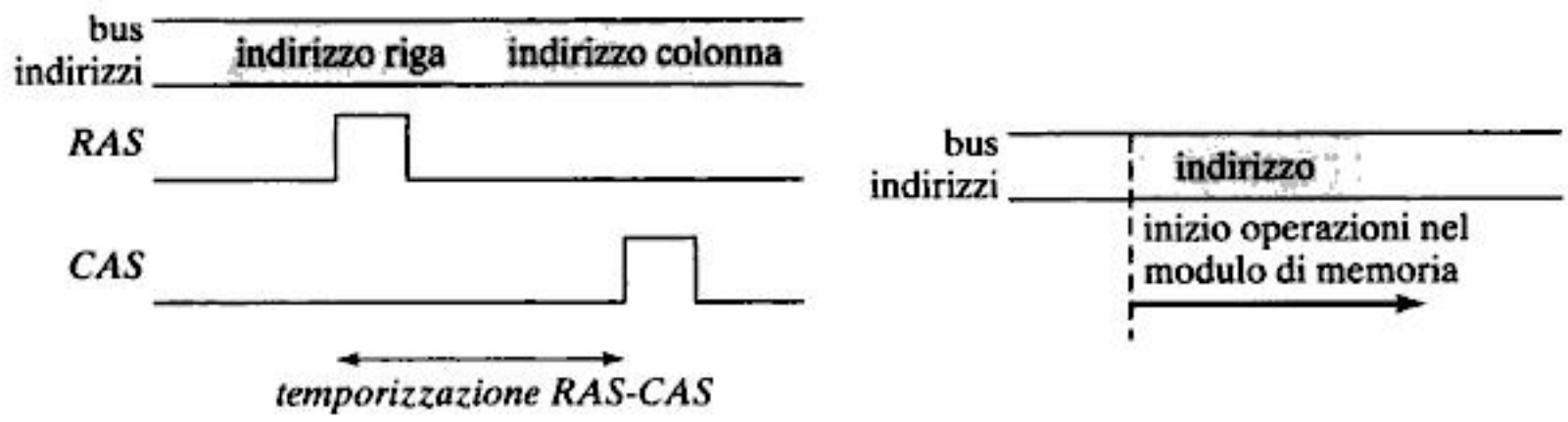
Infine, altri due componenti della memoria spesso trascurati sono l'interfaccia di *ingresso/uscita (I/O)* e *la circuiteria di controllo*. Il tipo di interfaccia I/O ha un'enorme importanza nel controllo e nella temporizzazione del modulo di memoria. Questa affermazione è meglio chiarita se si confrontano i tipici diagrammi temporali ingresso-uscita di una memoria DRAM e una memoria SRAM.

Fin dalle origini, i costruttori di memorie DRAM hanno scelto di fornire l'indirizzo della cella in due fasi temporali distinte. In base a questa strategia la metà superiore e quella inferiore dell'indirizzo sono trasmesse alla memoria attraverso lo stesso bus di indirizzi, riducendo il numero di piedini del package. Questo tipo di approccio è poi sopravvissuto anche nelle generazioni seguenti. Infatti, il volume di produzione delle memorie DRAM è generalmente molto elevato e diminuire il numero di piedini significa ridurre le dimensioni e il costo, anche se a spese delle prestazioni. La presenza di un nuovo indirizzo all'ingresso del bus è confermata mediante l'uso di appositi segnali di sincronismo (vedi Figura 12-8a.) Abilitando il segnale *RAS (row-access strobe)* si comunica alla memoria che la metà più significativa della parola di indirizzo è presente all'ingresso del bus indirizzi e che può iniziare l'operazione di decodifica della riga. La parte meno significativa dell'indirizzo è fornita in seguito, quando il segnale *CAS (column-access strobe)* è abilitato. Per garantire il funzionamento corretto del modulo di memoria è necessario scegliere accuratamente l'intervallo di tempo tra i segnali *RAS* e *CAS*. Infatti, i segnali *RAS* e *CAS* costituiscono i segnali di clock del modulo di memoria e sono usati per sincronizzare operazioni come la decodifica, l'accesso alla matrice di celle e la gestione delle periferiche di uscita.

Per le memorie SRAM, al contrario, si è scelto un approccio *autotemporizzato*, come illustrato nella Figura 12-8b. La parola corrispondente all'indirizzo della cella è fornita all'ingresso in un solo colpo e un'apposita circuiteria interna rileva qualsiasi variazione dei

[2]La parola con indirizzo più alto corrisponde generalmente all'ultimo dato inserito in una memoria cache, che costituisce la corrispondenza preferenziale.

Materiale protetto da copyright



(a) Temporizzazione di una DRAM (b) Temporizzazione di una SRAM

Figura 12-8 Interfaccia I/O delle memorie DRAM e SRAM e relativi segnali di controllo.

bit del bus indirizzi. Non è richiesto alcun segnale di temporizzazione esterno, ma tutti gli eventi interni (abilitazione dei decodificatori o dei sense amplifier) sono gestiti mediante segnali generati internamente. Questo tipo di approccio fa sì che il tempo di ciclo di una memoria SRAM sia molto simile o addirittura uguale al corrispondente tempo di accesso, cosa che non accade nelle memorie DRAM. L'aumento complessivo delle prestazioni degli elaboratori che impiegano memorie DRAM come componente per l'immagazzinamento delle informazioni, obbliga i costruttori a mantenere la velocità delle DRAM al passo con quella dei microprocessori. Sono stati proposti diversi approcci per migliorare le prestazioni delle memorie DRAM. Alcuni esempi sono le DRAM sincrone *(SDRAM, Synchronous DRAM)* e le *Rambus DRAM* (RDRAM). La novità principale in questo tipo di architetture, che saranno esaminate più avanti, non è nella struttura delle celle che compongono il nucleo della memoria, ma nel modo con cui comunicano con il mondo esterno.

Progettare la circuiteria di controllo e temporizzazione in modo tale che la memoria conservi la sua funzionalità nonostante l'ampio spettro di variazioni nei processi di fabbricazione e temperature operative è un compito molto impegnativo che richiede simulazioni e ottimizzazione a tutti i livelli dei progettazione. Tutto ciò è parte integrante, ma spesso trascurata, del lavoro di progettazione di una memoria e ha un'influenza fondamentale sull'affidabilità, sulla robustezza e sulle prestazioni del modulo di memoria.

12.2 Il nucleo della memoria: la matrice di celle

Questo paragrafo è dedicato al progetto del nucleo della memoria, cioè la matrice di celle, per vari tipi di memorie a semiconduttore. Sebbene il problema fondamentale nel progetto di grandi banchi di memoria sia quello di mantenere piccole le dimensioni della cella, tale obiettivo deve essere perseguito in modo tale che le altre figure di merito come la robustezza e la velocità non siano fatalmente compromesse. Saranno trattate nell'ordine le memorie *a sola lettura, non volatili e a lettura-scrittura.* A conclusione saranno brevemente discusse le memorie associative.

12.2.1 Memorie a sola lettura

Sebbene l'idea stessa di una memoria che può essere solo letta e mai modificata possa sembrare strana a prima vista, una riflessione più approfondita rivela molte applicazioni poten-

Materiale protetto da copyright

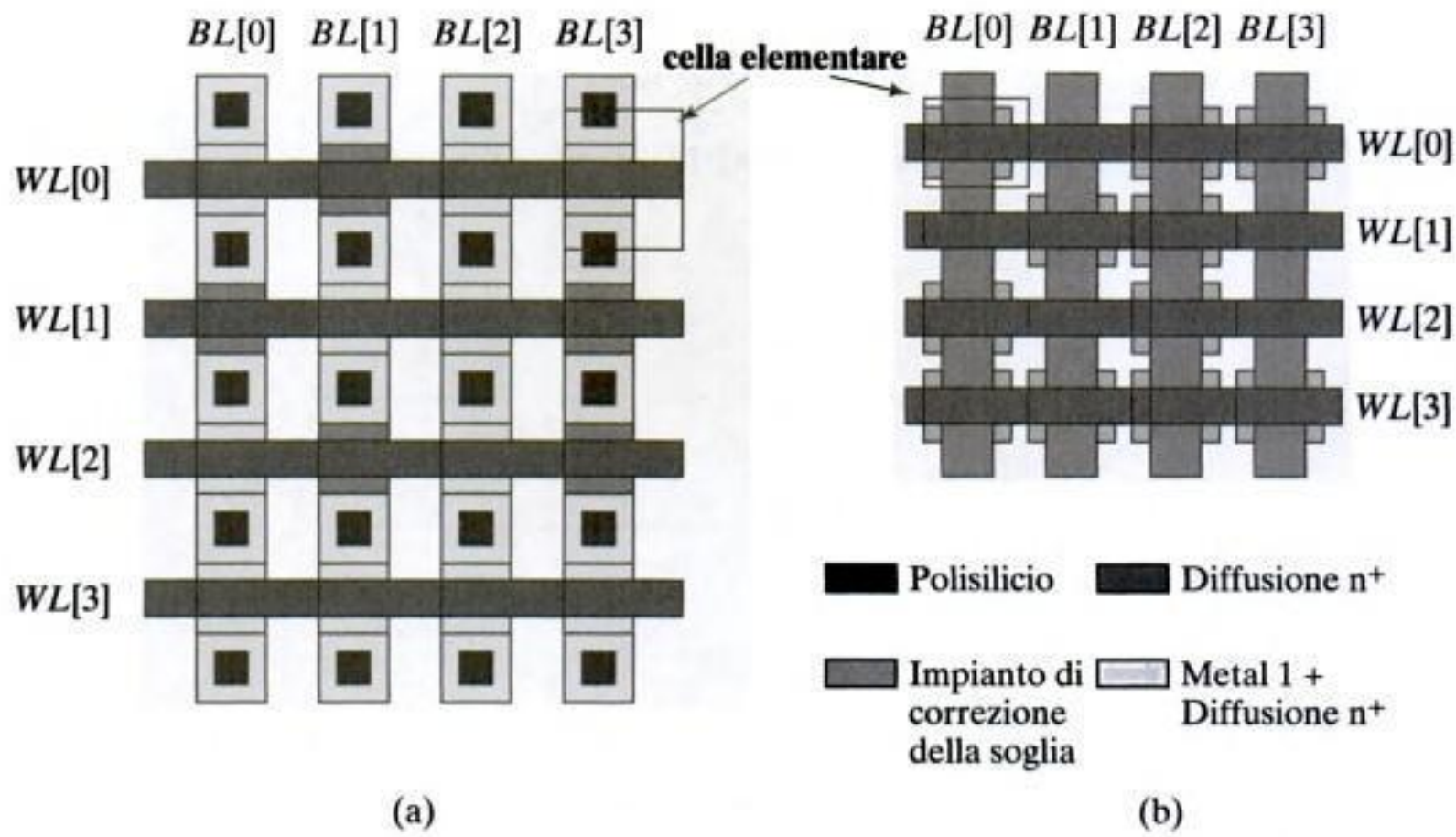


Figura 12-14 Layout di una memoria ROM-NAND con 4 × 4 celle: (a) programmazione delle celle mediante la maschera METAL-1 (area: 8 λ × 7 λ); (b) programmazione mediante la tensione di soglia (area: 5 λ × 6 λ), assumendo che l'impianto per regolare la tensione di soglia debba estendersi per 1 λ in tutte le direzioni attorno al gate del transistor.

drain di un transistor (a). La cella risultante ha un'area del 15% circa minore della più piccola cella ROM-NOR. Una riduzione ancora maggiore dell'area della cella può essere ottenuta se fosse a disposizione un ulteriore passo di processo che permette di variare la tensione di soglia dei transistor in modo selettivo per programmare le celle di memoria. La tensione di soglia del transistor può essere abbassata introducendo impurità di tipo n nel substrato del transistor (impianto per la correzione della tensione di soglia). Se la tensione di soglia è abbassata a valori negativi, il transistor diventa a svuotamento ed è acceso anche con tensione gate-source nulla. Questo approccio di fatto equivale a cortocircuitare il source e il drain. La cella ottenuta ha un'area pari a 30 λ^2, meno di metà della cella ROM-NOR equivalente. Tuttavia, esiste un prezzo da pagare per ottenere una tale densità di integrazione: nei prossimi paragrafi dimostreremo che la configurazione a NAND comporta una notevole perdita di prestazioni ed è utile per matrici di memorie molto piccole. Inoltre, l'impianto di correzione della soglia richiede un'ulteriore maschera e un passo di processo aggiuntivo che non è sempre disponibile al progettista.

Esempio 12.3 Escursione logica nelle memorie ROM con struttura NAND e NOR

Supponendo che i due layout riportati nelle Figure 12-12 e 12-14 siano disegnati usando la nostra tecnologia CMOS di riferimento da 0.25 μm, calcoliamo le dimensioni dei transistor PMOS del pull-up in modo tale che il valore di tensione di livello logico basso V_{OL} sia, nel caso peggiore, minore di 1.5 V (considerando una tensione di alimentazione di 2.5 V). Con questi valori otteniamo un'escursione logica della bitline

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Prestazioni dinamiche delle memorie ROM

La risposta dinamica di una matrice di memoria è definita come il tempo trascorso dalla commutazione della wordline all'istante in cui la bitline compie un escursione di tensione con una data ampiezza ΔV. Poiché l'uscita della bitline è generalmente connessa ad un *sense amplifier*, non è necessario che la sua escursione sia completa. Il sense amplifier è in grado di rilevare anche piccole variazioni della tensione della bitline. Valori tipici di escursione ΔV sono intorno a 0.5 V.

La differenza più importante tra l'analisi del tempo di ritardo di una porta logica e quello di una memoria è che in questo secondo caso la maggior parte del ritardo è attribuibile agli elementi parassiti delle piste di interconnessione. Quindi, è esternamente importante realizzare un modello accurato e adeguato di questi elementi parassiti. Ciò è illustrato nell'esempio che segue, che mostra come possiamo estrarre i valori di capacità e resistenza parassita delle piste di wordline e bitline in entrambe le strutture della matrice (NOR e NAND) introdotte precedentemente. Tratteremo questo esempio con estrema cura, poiché questo tipo di analisi è valida per altri tipi di memorie, come le SRAM e le DRAM.

Esempio 12.4 Parametri parassiti delle piste di wordline e bitline

In primo luogo è necessario ricavare un modello equivalente delle celle di memoria per la strutture NOR e NAND riportate rispettivamente nella Figura 12-12b e nella Figura 12-14b. Tenere in conto tutti i transistor della matrice condurrebbe presto ad un'analisi complessa con equazioni e modelli intrattabili, specialmente in memorie di grandi dimensioni. Semplicità e astrazione sono invece le vie più ovvie da seguire. Nel seguito considereremo il caso di una matrice di memoria con 512×512 celle.

1. ROM-NOR

La Figura 12-15 introduce un modello adatto all'analisi del tempo di propagazione lungo le wordline e le bitline di una memoria ROM-NOR. La wordline è schematizzabile come una linea *RC* a parametri distribuiti, poiché è realizzata con una pista di polisilicio con una resistenza superficiale abbastanza elevata. Per diminuire la resistenza della wordline e migliorare le proprietà dinamiche è consigliabile depositare del saliciuro sopra la pista di polisilicio. Al contrario la bitline è realizzata in alluminio, e la resistenza della pista entra in gioco solo per interconnessioni molto lunghe. In condi-

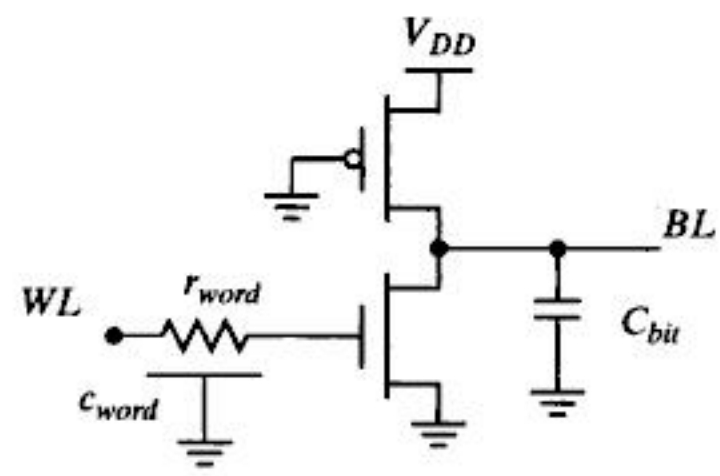


Figura 12-15 Modello dinamico equivalente di una ROM-NOR.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

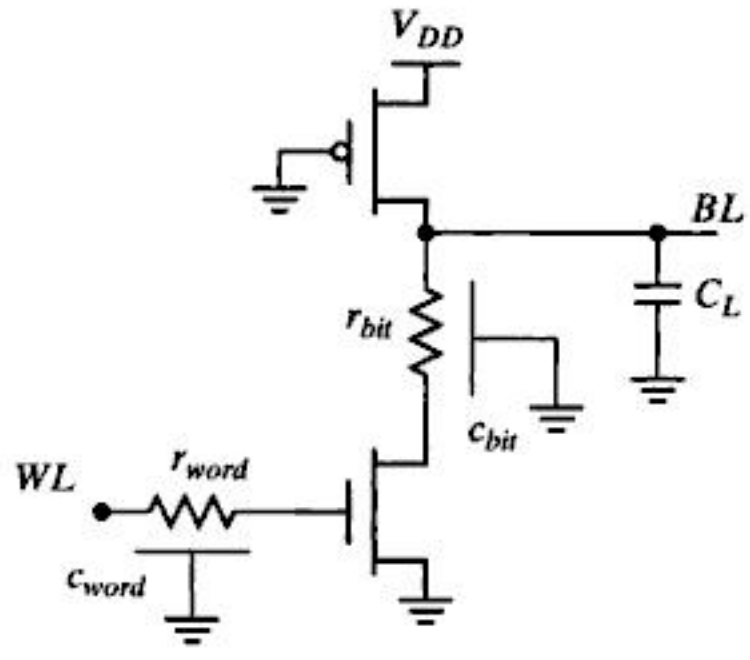


Figura 12-16 Modello equivalente per le wordline e le bitline di una memoria ROM-NAND

In memorie di grandi dimensioni il modello che meglio descrive l'intera catena di transistor è una rete *RC* a parametri distribuiti.

I parametri parassiti della wordline (per la matrice di memoria in Figura 12-14) sono i seguenti:

Resistenza per cella: $(6/2) \times 5\ \Omega/\text{sq} = 15\ \Omega$

Capacità della pista per cella:

$$(3\lambda \times 2\lambda)\,(0.125)^2\,0.088 + 2 \times (3\lambda \times 0.125) \times 0.054 = 0.049\ \text{fF}$$

Capacità di gate per cella: $(3\lambda \times 2\lambda)\,(0.125)^2\,6 = 0.56\ \text{fF}$

I parametri parassiti della bitline sono i seguenti:

Resistenza per cella: $13 / 1.5 = 8.7\ \text{k}\Omega$

Capacità della pista per cella: incluse nella capacità di source/drain dei transistor

Infine, la capacità di source/drain per cella è data da:

$$((3\lambda\times3\lambda)\,(0.125)^2\times2\times0.56+2\times3\lambda\times0.125\times0.28\times0.6+(3\lambda\times2\lambda)\,(0.125)^2\times6 = 0.85\ \text{fF}$$

La capacità di source (drain) deve includere anche il contributo gate-source (gate-drain), cioè l'intera capacità di gate. Questo è in contrasto con la configurazione a NOR, nella quale l'unico contributo deriva dalla capacità di sovrapposizione gate-drain e gate-source.

Il calcolo della resistenza media dei transistor è più complesso. Infatti, il valore della resistenza cambia da transistor a transistor in funzione della sua posizione lungo la catena, a causa della differente tensione gate-source e dell'*effetto body*. In questa analisi al primo ordine, possiamo usare il modello semplificato per la resistenza equivalente introdotto nel Capitolo 3. Questa approssimazione fornisce ugualmente risultati ragionevoli nel caso della porta logica NAND a 4 ingressi analizzata nel Capitolo 6 (Esempio 6.4).

Materiale protetto da copyright

Ci si potrebbe chiedere perché dobbiamo annoiarci con questi modelli, dato che le simulazioni ci forniscono senza difficoltà risultati più accurati. Dobbiamo prestare attenzione al fatto che un modulo di memoria potrebbe contenere da qualche migliaio a qualche milione di transistor, rendendo proibitivamente lente le ripetute simulazioni. Il modello semplificato ci aiuta anche a capire senza molte difficoltà il comportamento della matrice di celle che forma il nucleo interno della memoria. Le simulazioni fatte al computer non consentono di identificare nè dove sono i colli di bottiglia del sistema nè il modo di trattarli.

A partire dagli elementi parassiti appena calcolati e i relativi modelli, possiamo ora calcolare il tempo di propagazione attraverso il nucleo della memoria.

Esempio 12.5 Tempo di ritardo di una memoria ROM-NOR con 512 × 512 celle

1. Il tempo di ritardo massimo lungo la pista di wordline che attraversa M celle può essere calcolato utilizzando il modello RC a costanti distribuite e usando l'espressione ricavata nel Capitolo 4:

$$t_{word} = 0.38\ (r_{word} \times c_{word})\ M^2 = 0.38\ (17.5\ \Omega \times (0.049 + 0.75)\ \text{fF})\ 512^2 = 1.4\ \text{ns}$$

2. Il tempo di ritardo lungo la bitline dipende dal verso della transizione. Assumendo che le dimensioni del transistor NMOS di pull-down siano 0.5/0.25 e quelle del dispositivo di pull-up siano 1.3125/0.25, come abbiamo trovato nell'Esempio 12.3, siamo in grado di calcolare il tempo di propagazione usando la tecnica a noi più familiare:

$$C_{bit} = 512 \times (0.8 + 0.09)\ \text{fF} = 0.46\ \text{pF}$$

$$t_{pHL} = 0.69\ (13\ \text{k}\Omega / 2 \parallel 31\ \text{k}\Omega / 5.25)\ 0.46\ \text{pF} = 0.98\ \text{ns}$$

Per il calcolo del tempo di propagazione relativo alla commutazione basso-alto usiamo un approccio simile:

$$t_{pLH} = 0.69\ (31\ \text{k}\Omega / 5.25)\ 0.46\ \text{pF} = 1.87\ \text{ns}$$

Uno sguardo ai risultati appena trovati ci rivela che il tempo di propagazione attraverso la wordline domina sul tempo di ritardo lungo la bitline. Infatti, il primo è dovuto quasi interamente alla resistenza della pista di polisilicio. Alcune delle tecniche per ridurre il ritardo della linea RC a parametri distribuiti, come abbiamo visto nel Capitolo 9, possono tornarci utili in questa sede. Pilotare le wordline da entrambi i lati o inserire delle piste metalliche parallele alla wordline stessa che la contattano in più punti (dette anche *wordline globali*) sono tecniche orientate alla soluzione del problema della resistenza parassita delle piste di polisilicio (Figura 9-18). Tuttavia, l'approccio più efficiente è quello di partizionare attentamente e con molta cura la memoria in sottoblocchi di dimensioni adeguate in modo da bilanciare i tempi di ritardo lungo le wordline e le bitline. La ripartizione della memoria aiuta anche a ridurre il consumo di energia associato alle commutazioni delle wordline e alla circuiteria che pilota le wordline stesse.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

Problema 12.4 ROM-NAND a precarica

Sebbene la logica a precarica si presti bene alla realizzazione di una memoria ROM-NOR, nascono alcuni problemi quando applichiamo la stessa struttura alla ROM-NAND. Spiegare perché.

Le eccellenti caratteristiche dell'approccio a precarica lo hanno reso la struttura attualmente più in voga. Di fatto tutte le memorie di grandi dimensioni, incluse le NVRWM e le RAM, sono attualmente realizzate in logica dinamica a precarica.

Applicazioni delle memorie ROM

Il lettore deve essere consapevole del fatto che la maggior parte dei problemi, sollevati nei paragrafi precedenti riguardo alle caratteristiche statiche, alla risposta dinamica o al consumo di potenza, hanno natura generale e riguardano anche altri tipi di memorie. Prima di affrontare la trattazione di queste altre strutture, vale la pena di spendere qualche parola sulla classificazione dei moduli ROM e sulle relative tecniche di programmazione. Una prima classe di moduli ROM comprende le cosiddette *memorie orientate ad applicazioni specifiche*, dove la memoria è parte integrante di un sistema più grande, finalizzato ad una particolare applicazione (*custom*), dove il modulo di memoria è programmato per quel particolare utilizzo. In queste condizioni il progettista ha tutti i gradi di libertà possibili ed è in grado di usare qualsiasi livello di maschera (o combinazioni di queste) per programmare il dispositivo.

Una seconda categoria è costituita dai cosiddetti *prodotti base* o *prodotti di uso generico*, cioè i chip di memoria ROM che il costruttore produce in massa e che sono poi personalizzati in accordo alle specifiche richieste del cliente. In questo caso è essenziale che il numero di passi di processo necessari alla fase di programmazione sia minimo e che sia possibile realizzarli nelle ultime fasi di fabbricazione del prodotto, in modo da poter produrre un'elevata quantità di chip non programmati, ma quasi completamente finiti. Questo tipo di programmazione *eseguito mediante una sola maschera* usa preferenzialmente gli ultimi livelli di layout (contatti, via e piste metalliche) per personalizzare e programmare la memoria, come è stato accennato negli esempi precedenti. La programmazione di un modulo di memoria ROM è responsabilità del costruttore, introducendo uno sgradito ritardo nello sviluppo del prodotto. Ciò ha fatto sì che, nel tempo, questo approccio sia divenuto sempre meno popolare. Recentemente una variante di questo metodo sta emergendo nell'arena della progettazione *system-on-a-chip*, dove la maggior parte del sistema è prefabbricato nel chip e solo una piccola porzione del chip è programmabile dal cliente, utilizzando preferibilmente i livelli superiori di metallizzazioni e contatti. Ad esempio, questo approccio può essere usato per programmare il microcontrollore, incluso nel chip, per una varietà di applicazioni.

Un approccio ancor più attraente per il cliente è quello che gli consente di programmare la memoria *in casa*. Una tecnologia che offre questa possibilità è la struttura PROM (Programmable ROM) che permette al cliente di programmare la memoria una sola volta. Questo tipo di programmazione è spesso realizzata inserendo nella cella di memoria dei *fusibili*, costruiti in nichrome (NiCr), polisilicio o altri conduttori. Durante la fase di programma-

Materiale protetto da copyright

zione, alcuni di questi fusibili sono fatti saltare, forzando una corrente elevata, eliminando in questo modo la connessione ai transistor.

Sebbene le memorie PROM abbiano il vantaggio di essere programmabili dal cliente, la possibilità di essere scritte una sola volta le rende poco attraenti. Ad esempio, un semplice errore nel processo di programmazione rende inutilizzabile il dispositivo. Da ciò deriva l'attuale tendenza verso quei componenti che possono essere programmati diverse volte (seppur lentamente). Il prossimo paragrafo tratta di questo tipo di memorie.

12.2.2 Memorie non volatili a lettura-scrittura

L'architettura delle memorie non volatili a lettura-scrittura NVRWM (non-volatile read-write memory) è di fatto identica alla struttura delle memorie ROM. Il nucleo della memoria è costituito da una matrice di transistor disposti in modo tale da formare una griglia di wordline e bitline. La memoria viene programmata disabilitando o abilitando selettivamente alcuni di questi dispositivi. Mentre in una memoria ROM questa operazione viene fatta a livello di layout programmando la maschera, in una memoria NVRWM è usato invece un transistor modificato che permette di variare la sua tensione di soglia elettricamente. La variazione della tensione di soglia è mantenuta indefinitamente (o almeno per un tempo paragonabile alla vita operativa del dispositivo) anche quando la tensione di alimentazione viene meno. Per riprogrammare la memoria, la cella deve essere prima *cancellata* e poi deve essere eseguito un nuovo ciclo di programmazione. Il metodo impiegato per cancellare la memoria è il fattore che differenzia i vari tipi di memorie non volatili riprogrammabili. Il tempo richiesto alla programmazione è almeno un ordine di grandezza maggiore del tempo di accesso durante l'operazione di lettura.

Iniziamo questo paragrafo introducendo il *transistor a gate flottante*, cioè il dispositivo che costituisce il nucleo della maggior parte delle memorie non volatili riprogrammabili. Il resto del paragrafo è invece dedicato alle diverse varianti di questo elemento, che riguardano soprattutto la procedura di cancellazione e che stanno alla base della classificazione delle famiglie di memorie non volatili. Il paragrafo si conclude con una breve discussione sulle nuove famiglie emergenti di memorie non volatili.

Il transistor a gate flottante

Nel corso degli anni sono stati fatti molti tentativi per realizzare un dispositivo le cui caratteristiche siano alterabili elettricamente e abbastanza affidabile da poter essere programmato molte volte. Ad esempio, il transistor MNOS *(metal nitride oxide semiconductor)* fu un tempo molto promettente, ma non ha avuto successo fino ad ora. In questo dispositivo la tensione di soglia viene modificata intrappolando elettroni in uno strato di nitruro di silicio (Si_3N_4) depositato sopra l'ossido di gate (SiO_2) [Chang77]. Una soluzione alternativa che ha da tempo riscosso molto successo è il transistor a gate flottante mostrato nella Figura 12-18, che costituisce l'elemento fondamentale di tutte le memorie non volatili costruite oggi. La struttura è simile a quella di un transistor MOS convenzionale, eccezione fatta per uno strato addizionale di polisilicio che viene inserito tra il gate e il canale. Questo strato di polisilicio è completamente isolato da tutti gli altri terminali e per questo motivo è detto *gate flottante*

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

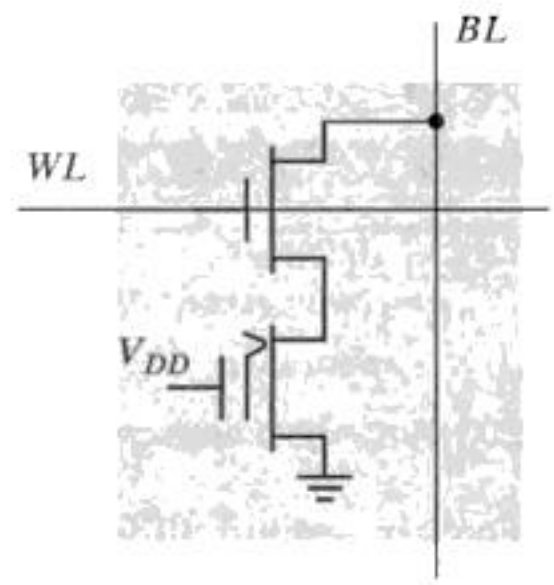


Figura 12-22 Cella EEPROM durante l'operazione di lettura. Se programmata, la tensione di soglia del dispositivo FLOTOX è maggiore di V_{DD}, disabilitando il dispositivo. Se cancellata (e non riprogrammata), si comporta come un interruttore chiuso.

nel Fowler–Nordheim [Snow67]. Le caratteristiche tensione corrente della giunzione silicio-ossido-polisilicio attraverso l'ossido di tunnel sono mostrate nella Figura 12-21b.

Il vantaggio principale di questo tipo di programmazione è costituito dal fatto che è reversibile, cioè la cancellazione è semplicemente eseguita invertendo la tensione applicata durante il processo di scrittura. Iniettare elettroni nel gate flottante significa aumentare la tensione di soglia, mentre estrarre elettroni dal gate flottante equivale a diminuire la tensione di soglia V_T. Questa bidirezionalità nella traslazione di V_T introduce, però, un problema nel controllo della tensione di soglia: rimuovendo troppa carica dal gate flottante è possibile che il dispositivo diventi un transistor a svuotamento, cioè con tensione di soglia negativa, che non può essere spento neanche quando la wordline è mantenuta a 0 V. Notiamo anche che la tensione di soglia risultante dal processo di programmazione dipende: dalla quantità iniziale di carica contenuta nel gate flottante; dalla tensione di programmazione; dallo spessore del sottile ossido di tunnel, soggetto a variazioni non trascurabili da un punto all'altro del wafer. Per porre rimedio a questo problema, nella cella EEPROM deve essere inserito un transistor in serie al dispositivo FLOTOX. Questo transistor ha la funzione di interruttore di accesso durante l'operazione di lettura, mentre al transistor FLOTOX è lasciato il compito di memorizzare l'informazione (vedi Figura 12-22). Ciò è in antitesi con la cella EPROM, dove il transistor FAMOS ha entrambe le funzioni di accesso e memorizzazione.

La cella EEPROM con i suoi due transistor risulta più grande della corrispondente cella EPROM. Questo costo in termini di area è ulteriormente aggravato dal fatto che la cella FLOTOX è intrinsecamente più grande della cella FAMOS a causa della presenza della zona di ossido di tunnel. Inoltre, la fabbricazione di uno strato di ossido molto sottile è un passo di processo difficile e costoso. Un banco di memoria EEPROM dunque integra meno bit ad un costo addirittura più alto rispetto alla memoria EPROM. Dal lato positivo della bilancia sta però la maggiore versatilità e il tempo di vita più lungo (tollera fino a 10^5 cicli di cancellazione e riprogrammazione).[4] I ripetuti cicli di programmazione causano la deviazione della

[4]L'ossido sottile non è l'unico modo per consentire l'operazione di tunnel. Altri approcci impiegano strutture con superfici sagomate in modo da aumentare il campo elettrico nei punti desiderati [Masuoka91].

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

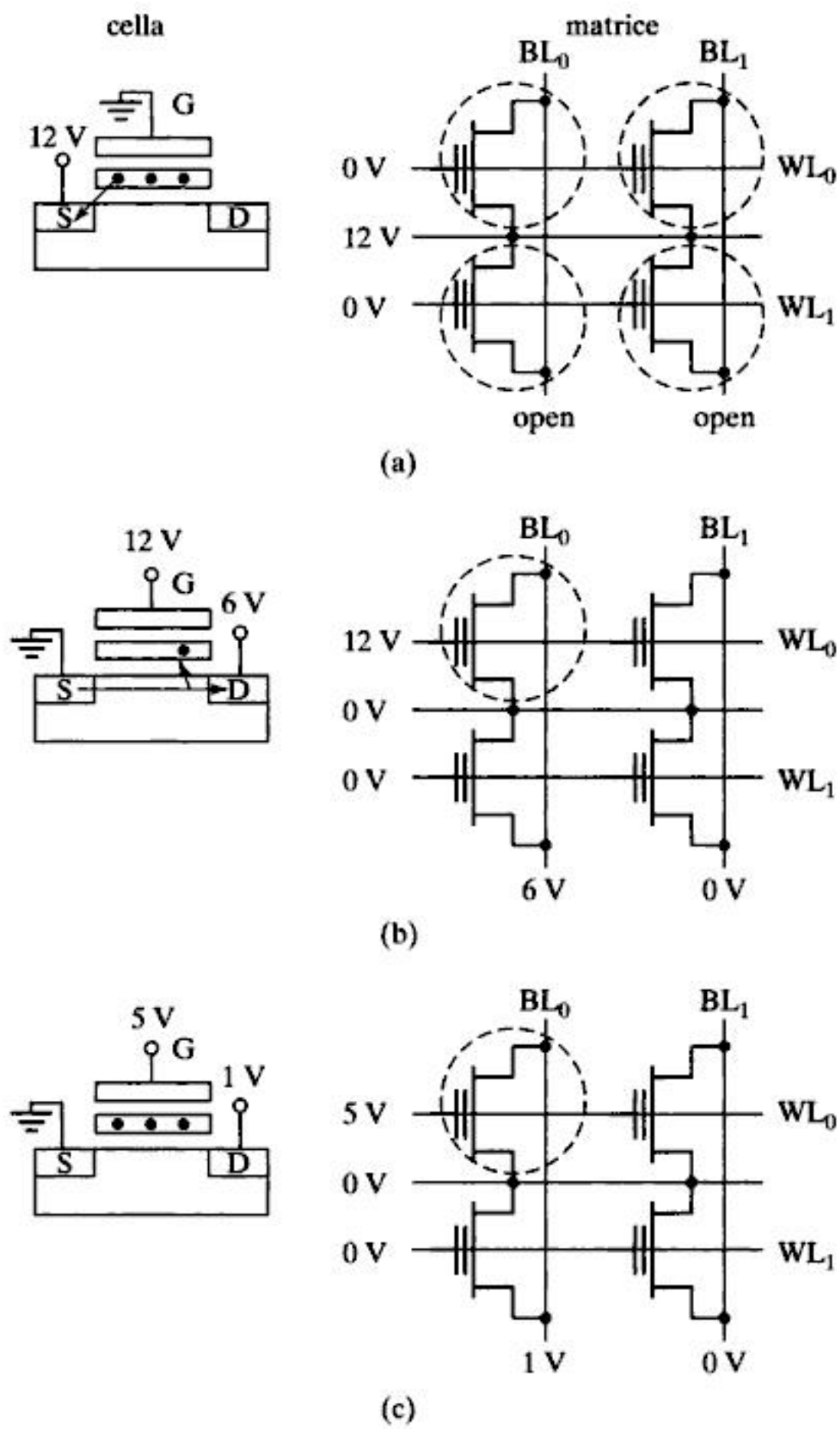


Figura 12-24 Operazioni fondamentali in una cella di memoria Flash-NOR [da Itoh01]: (a) cancellazione; (b) programmazione; (c) lettura.

zione deve iniziare con l'*operazione di cancellazione* (a): il gate è mantenuto a 0 V e il source a una tensione elevata (12 V); gli elettroni del floating gate, se presenti, sono iniettati verso il source (nodo a potenziale maggiore) per effetto tunnel. Tutte le celle sono cancellate simultaneamente. Il diverso valore iniziale della tensione di soglia delle celle, così come la variazione statistica dello spessore dell'ossido di tunnel tra una cella e l'altra, potrebbe causare una certa dispersione dei valori di tensione di soglia delle celle al termine della fase di cancellazione. Per rimediare al problema si deve procedere nel modo seguente: (1) prima di iniziare la cancellazione, tutte le celle nel blocco sono programmate in modo che abbiano

Materiale protetto da copyright

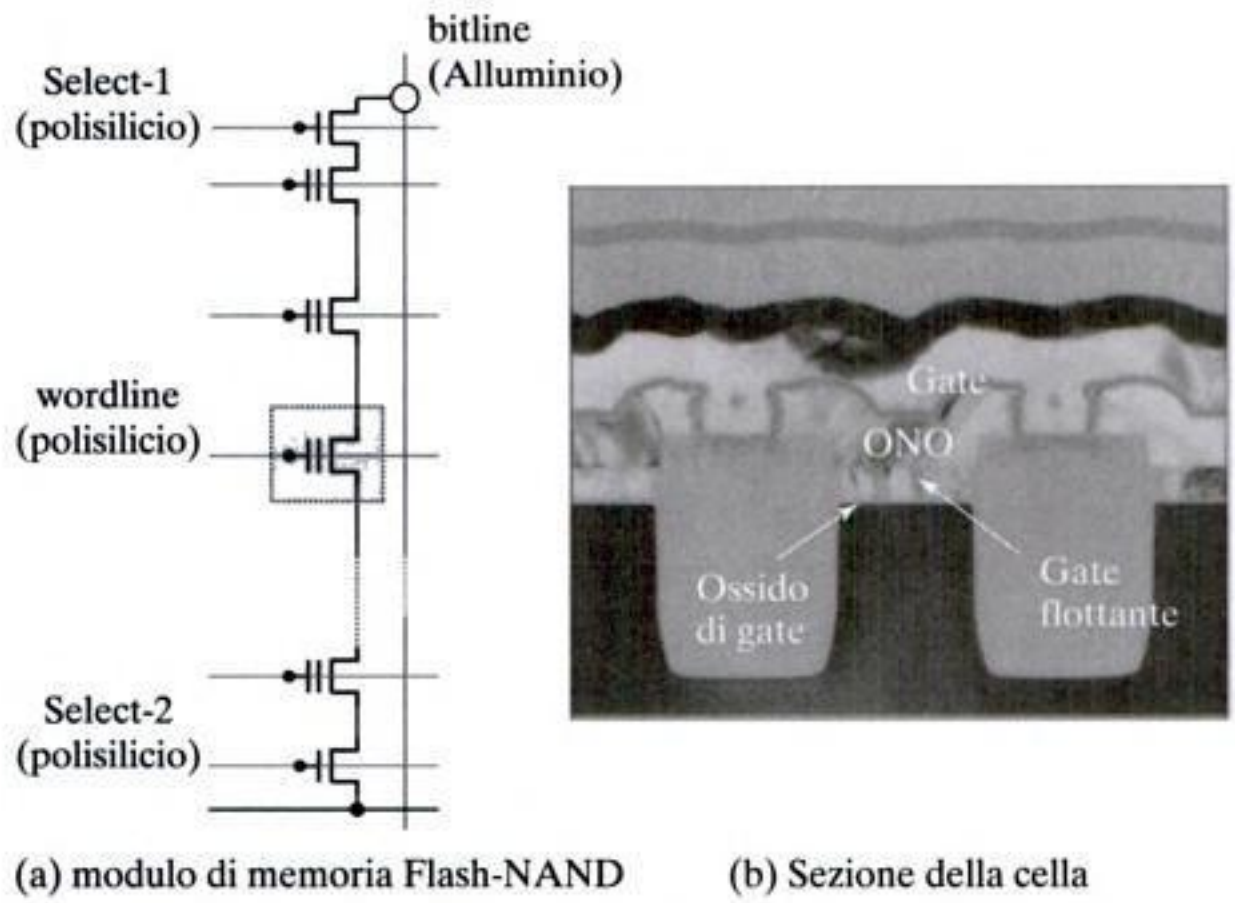


Figura 12-25 Modulo di memoria Flash con architettura NAND [Nakamura02]. Il gate di controllo impiega un isolante con elevata costante dielettrica (ONO, Ossido-Nitruro-Ossido) per aumentare la capacità C_{FG} e quindi l'effetto dell'iniezione di carica sulla tensione di soglia.

avendo abilitato entrambi i transistor di selezione. L'impiego dell'effetto tunnel come meccanismo di scrittura e cancellazione riduce la corrente necessaria a tali operazioni e consente la programmazione in parallelo di molti moduli di memoria, pur mantenendo sotto controllo il consumo di potenza. Il tempo di programmazione per byte è circa 200-400 ns. La possibilità di riprogrammare in parallelo l'intero blocco di memoria, permette di eseguire la cancellazione in modo veloce e affidabile, consentedo la cancellazione dell'intero chip in circa 100 ms.

Per raggiungere il miglior compromesso tra tempo di accesso, velocità di cancellazione e programmazione e densità di integrazione, sono state proposte numerose architetture alternative per le memorie Flash, il cui funzionamento si basa sempre sui principi appena esposti. Per una descrizione più dettagliata, invitiamo il lettore a consultare lavori come [Itoh01]. Una questione merita, però, un'attenzione speciale: tutte le memorie non volatili hanno bisogno di una tensione di alimentazione molto elevata (> 10 V) per eseguire le operazioni di cancellazione e programmazione. Alcune di queste tensioni possono essere generate all'interno del chip usando circuiti a pompa di carica, specialmente se la corrente assorbita dalla cella è piccola, come nel caso del tunnel Fowler-Nordheim. Al contrario, l'iniezione di elettroni caldi richiede una corrente elevata (> 0.5 mA) e rende necessario l'impiego di una seconda tensione di alimentazione esterna, aggravando i costi e la complessità del sistema. Ovviamente, la tendenza è quella di muoversi verso le memorie non volatili che non richiedono nè tensioni troppo elevate nè ulteriori fonti di alimentazione esterne, generando internamente tutte le tensioni necessarie.

Materiale protetto da copyright

Nuovi sviluppi nel mercato delle memorie non volatili

Tutte le memorie non volatili discusse finora basano il loro funzionamento sul transistor a gate flottante. Nel corso degli anni, sono stati proposti differenti approcci per realizzare l'immagazzinamento non volatile dell'informazione. Sebbene molti di questi non abbiano avuto una grande fortuna e non siano mai entrati nel mercato, alcune celle con struttura innovativa stanno sempre più prendendo piede e potrebbero avere un grande impatto nell'immediato futuro. Tra queste, le due famiglie più importanti e promettenti sono le memorie ferroelettriche (*FRAM, Ferroelectric RAM*) e le memorie magnetoresistive (*MRAM, Magnetoresistive RAM*). Tuttavia, è prematuro affermare che il transistor a gate flottante si stia avvicinando alla fine del suo tempo: nuove tecniche, quali le celle di memoria multilivello forniscono la possibilità di memorizzare informazioni ad altissimi livelli di integrazione. Nel seguito discuteremo brevemente i concetti chiave di ciascuno di questi interessanti sviluppi delle memorie non volatili.

Memorie non volatili multilivello Tutti i tipi di celle di memorie considerate finora sono bistabili, cioè la cella può memorizzare un bit 0 o un bit 1. Sarebbe possibile incrementare sensibilmente la densità della memoria se una singola cella fosse in grado di memorizzare più di due stati. Il limite pratico a questo tipo di approccio è che richiede un elevato rapporto segnale-rumore nella cella di memoria: immagazzinando k bit in una singola cella, l'escursione massima del segnale si riduce di $1/(2^k - 1)$. Fortunatamente, la possibilità di regolare la tensione di soglia V_T singolarmente in ciascuna cella aiuta ad ottenere il margine di rumore richiesto. È stata già dimostrata la fattibilità di memorie non volatili commerciali a due bit per cella [Baur95], ma con la continua riduzione della tensione di alimentazione è altamente improbabile che le memorie non volatili multilivello possano avere un futuro redditizio.

FRAM Le *memorie ferroelettiche* sono entrate nel gioco da un po' di tempo. Dopo molti falsi inizi, sembra che le FRAM stiano finalmente soddisfacendo le aspettative. Contrariamente al "transistor programmabile" usato nelle attuali NVRWM, il funzionamento delle memorie FRAM è basato su un "condensatore programmabile". Il dielettrico del condensatore è costituito da un materiale con proprietà ferroelettriche appartenente alla famiglia della *Perovskite*. Quando viene applicato un campo elettrico sufficientemente elevato attraverso il materiale, i cristalli assumono una certa polarizzazione dielettrica, che è mantenuta anche dopo la rimozione del campo elettrico. Per depolarizzare il materiale è necessario applicare un campo elettrico di verso opposto.

Una cella di memoria costruita a partire da questo principio, assomiglia alla struttura della cella DRAM a un transistor, che sarà discussa nei paragrafi seguenti. Durante l'operazione di scrittura, il condensatore è polarizzato in un verso o nell'altro. Per leggere il contenuto della cella, viene applicato un campo elettrico esterno. A seconda dello stato di polarizzazione del condensatore, può essere o meno generata una corrente che è poi letta dal sense amplifier.

Il vantaggio della memoria FRAM è una densità di integrazione elevata, che porta ad una cella di dimensioni inferiori a quelle della DRAM, ma con la caratteristica di conservare l'informazione anche in assenza della tensione di alimentazione. Il numero di cicli di

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

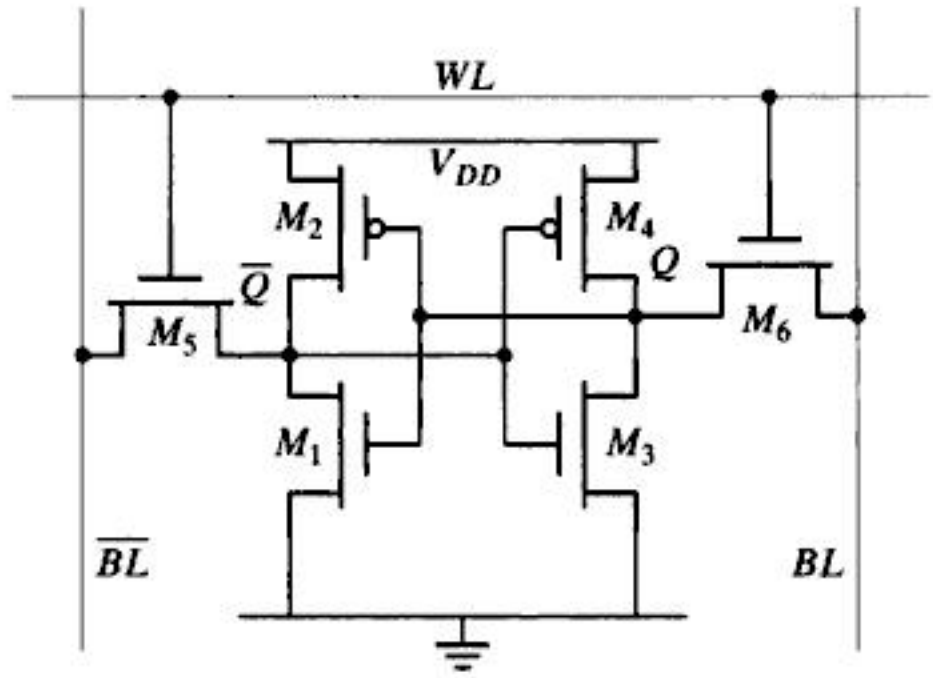


Figura 12-26 Cella di memoria SRAM a 6 transistor.

Funzionamento della cella SRAM La cella SRAM deve avere dimensioni più piccole possibile per raggiungere la massima densità di integrazione. Tuttavia, è necessario rispettare alcuni vincoli, per ottenere un funzionamento affidabile e robusto.

Per capire meglio il funzionamento di questa cella di memoria, consideriamo separatamente le operazioni di lettura e scrittura. Nel fare questo ricaviamo anche i vincoli da rispettare nel dimensionamento dei transistor.

Esempio 12.8 Cella SRAM: lettura

Supponiamo che la cella memorizzi un bit 1, cioè il nodo Q sia al valore logico 1. Ipotizziamo inoltre che entrambe le bitline siano precaricate alla tensione 2.5 V, prima che l'operazione di lettura abbia inizio. Il ciclo di lettura inizia quando la wordline viene portata al valore logico 1, abilitando entrambi i pass-transistor M_5 e M_6 con un certo tempo di ritardo pari al tempo di propagazione lungo la wordline. Se l'operazione di lettura è eseguita correttamente, i valori immagazzinati nei nodi Q e $\overline{Q}$ sono trasmessi verso le corrispondenti bitline, lasciando la BL al valore di precarica e scaricando la $\overline{BL}$ a massa attraverso la serie M_1–M_5. È necessario dimensionare attentamente i transistor per evitare che accidentalmente la cella commuti nel valore 0. Questo tipo di malfunzionamento è frequentemente detto *disturbo di lettura (read upset)*.

La situazione appena descritta è illustrata nella Figura 12-27. Consideriamo la parte del circuito dal lato della $\overline{BL}$. In memorie molto grandi la capacità associata a ciascuna delle bitline è molto elevata (alcuni pF); quindi, il valore della $\overline{BL}$ rimane costante al valore di precarica V_{DD} all'inizio dell'operazione di lettura, cioè quando la wordline commuta da 0 a 1. I due transistor NMOS in serie scaricano la $\overline{BL}$ a massa. Affinché l'area della cella sia piccola, i suoi transistor devono essere disegnati con dimensioni minime o quasi minime, conducendo ad elevati tempi di scarica della bitline a causa della elevata capacità. Non appena la differenza di potenziale tra la BL e la $\overline{BL}$ aumenta, entra in gioco il sense amplifier che accelera il processo di scarica e quindi l'intera operazione di lettura.

Materiale protetto da copyright

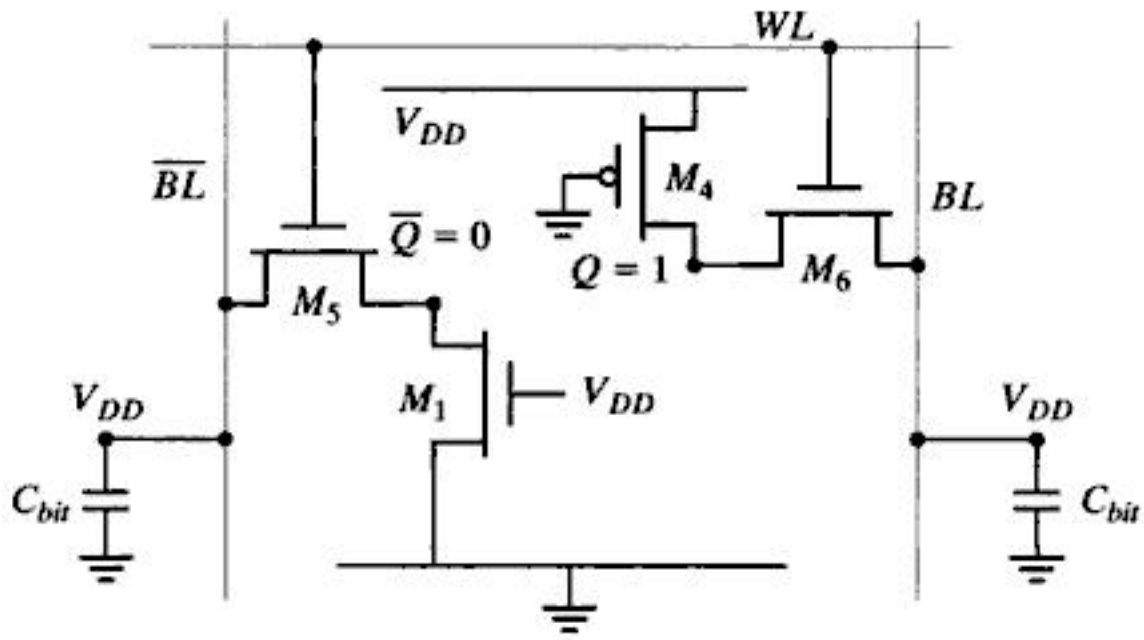


Figura 12-27 Modello semplificato di una cella SRAM durante l'operazione di lettura ($Q = 1$, $V_{precarica} = V_{DD}$).

Inizialmente, in corrispondenza del fronte di salita della *WL*, il potenziale del nodo intermedio tra i due transistor NMOS, $\overline{Q}$, aumenta verso il valore della tensione di precarica della $\overline{BL}$. La tensione massima raggiunta dal nodo $\overline{Q}$ deve rimanere abbastanza bassa in modo tale da non far scorrere una corrente troppo elevata attraverso l'invertitore M_3–M_4, che nel peggiore dei casi potrebbe provocare la commutazione indesiderata della cella. La resistenza del transistor M_5 deve essere sensibilmente maggiore di quella di M_1 per prevenire tale evento.

È possibile ottenere la relazione limite tra le dimensioni dei transistor uguagliando le correnti attraverso M_1 e M_5 con il massimo valore permesso per la variazione di tensione ΔV del nodo $\overline{Q}$. Per semplicità, trascuriamo l'effetto body sul transistor M_5 e otteniamo:

$$k_{n,M5}\left((V_{DD} - \Delta V - V_{Tn})V_{DSATn} - \frac{V_{DSATn}^2}{2}\right) = k_{n,M1}\left((V_{DD} - V_{Tn})\Delta V - \frac{\Delta V^2}{2}\right) \quad (12.2)$$

che fornisce l'espressione:

$$\Delta V = \frac{V_{DSATn} + CR(V_{DD} - V_{Tn}) - \sqrt{V_{DSATn}^2(1 + CR) + CR^2(V_{DD} - V_{Tn})^2}}{CR} \quad (12.3)$$

dove *CR* è detto *fattore di forma della cella (cell ratio)* ed è definito come:

$$CR = \frac{W_1/L_1}{W_5/L_5} \quad (12.4)$$

Il valore massimo dell'escursione di tensione ΔV in funzione di *CR* per una tecnologia CMOS da 0.25 μm è riportato nella Figura 12-28. Per evitare che la tensione del nodo intermedio salga sopra la tensione di soglia del transistor NMOS (circa 0.4 V), il fattore di forma della cella deve essere maggiore di 1.2. Per memorie di grandi dimensioni è preferibile mantenere le dimensioni della cella minime, pur garantendo la robustezza in fase di lettura. Se il transistor M_1 ha dimensioni minime, il pass-transistor di

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

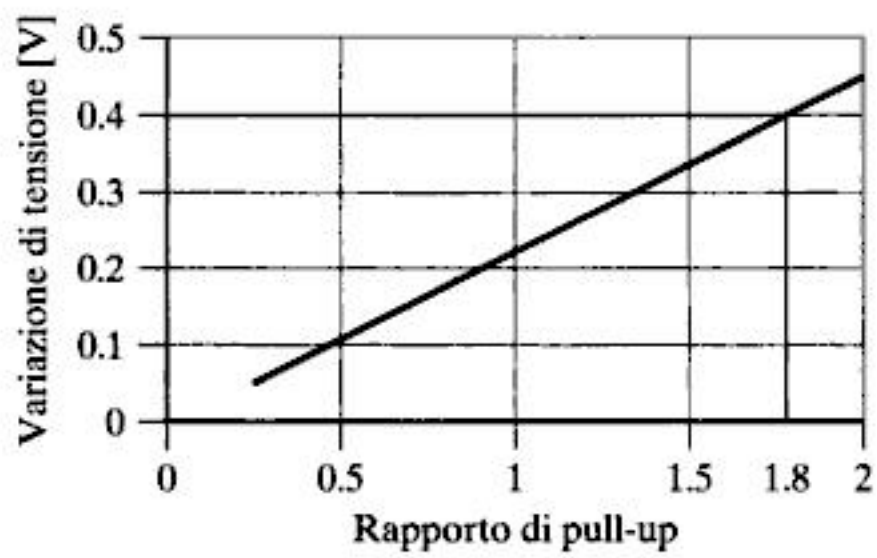


Figura 12-30 Tensione del nodo di uscita della cella in funzione del fattore di forma del pull-up della cella *PR* (rapporto tra i fattori di forma del transistor PMOS e del pass-transistor di accesso). *PR* deve essere inferiore a 1.8.

$$PR = \frac{W_4/L_4}{W_6/L_6}. \tag{12.7}$$

Il valore di V_Q da *PR* per una tecnologia CMOS da 0.25 μm è mostrato nella Figura 12-30: minore è PR, minore è il valore V_Q. Se desideriamo far scendere il potenziale del nodo Q sotto V_{Tn}, il *rapporto di pull-up* deve essere minore di 1.8.

Questa condizione limite è raggiunta, con buon margine, usando il fattore di forma minimo sia per il transistor PMOS M_4 sia per il pass-transistor NMOS M_6. Tuttavia, il progettista deve assicurare che la condizione di scrivibilità sia soddisfatta anche in presenza di dispersione nei parametri del processo di fabbricazione. Il caso peggiore si verifica quando il transistor PMOS è forte (più conduttivo), mentre il pass-transistor NMOS è debole (meno conduttivo) e la cella di memoria lavora con una tensione di alimentazione maggiore di quella nominale [Preston01].

La nostra ipotesi iniziale era che i transistor M_1 e M_2 non partecipino all'operazione di scrittura. Nella realtà questo non è completamente vero: appena uno dei due invertitori della cella inizia a commutare, il secondo lo segue, innestando la retroazione positiva.

Prestazione della cella SRAM Analizzando il comportamento dinamico di una cella SRAM ci si rende conto che l'operazione critica è la lettura. Essa richiede di caricare o scaricare l'elevata capacità delle bitline attraverso la serie dei due transistor molto piccoli della cella selezionata. Nell'esempio della Figura 12-27 la capacità $C_{\overline{BL}}$ deve essere scaricata attraverso la serie di M_5 e M_1. Il tempo di scrittura è dominato dal tempo di propagazione lungo la coppia di invertitori mutuamente retroazionati, dato che i circuiti che pilotano le due bitline (BL e $\overline{BL}$) possono essere realizzati anche con dimensioni abbastanza grandi. Per accelerare i tempi di lettura, le memorie SRAM fanno uso del sense amplifier: appena la differenza di potenziale tra BL e $\overline{BL}$ raggiunge un valore critico il sense amplifier viene attivato, scaricando una delle due bitline.

Ottimizzazione della cella SRAM La cella di memoria SRAM a sei transistor, sebbene semplice e affidabile, è caratterizzata da un notevole impiego di area. Oltre ai dispositivi,

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

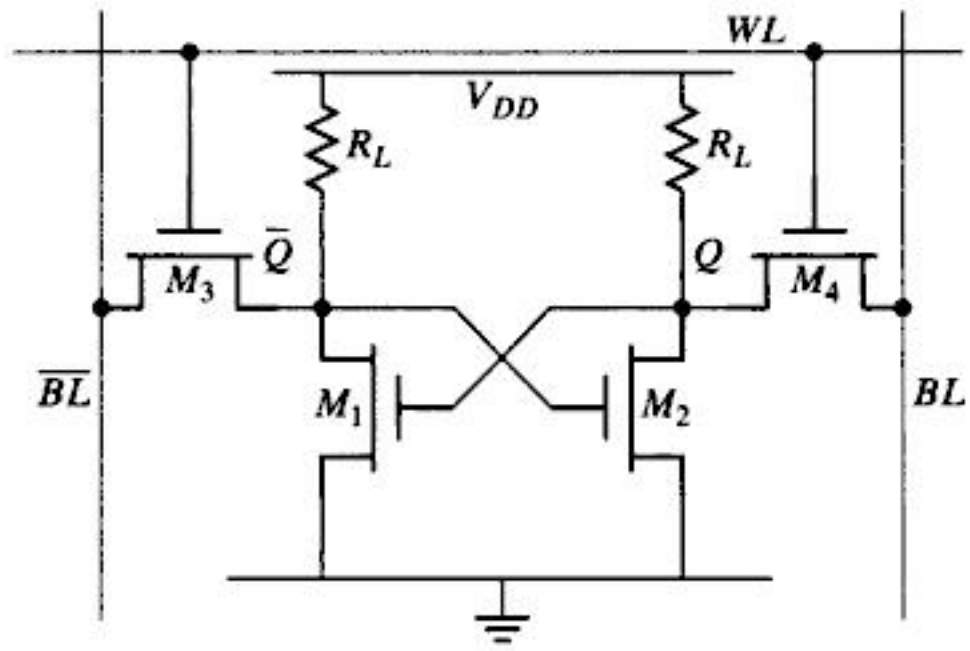


Figura 12-32 Cella SRAM con carico resistivo.

Consideriamo lo schema della cella mostrato nella Figura 12-32, detta cella *SRAM con carico resistivo* (e nota anche come cella SRAM a 4 transistor). La caratteristica fondamentale di questo tipo di cella è che i due invertitori CMOS sono sostituiti da una coppia di invertitori NMOS con carico resistivo. I transistor PMOS sono sostituiti da due resistenze, semplificando le interconnessioni e riducendo le dimensioni della cella di circa un terzo, come illustra la Tablella 12-2 per una memoria SRAM a 1 Mbit.

Analizzando la scrivibilità della cella SRAM, troviamo che esiste solo un limite inferiore al valore di resistenza del pull-up. Poiché le bitline sono precaricate esternamente, la cella non è coinvolta nel processo di carica e non esiste alcun svantaggio nel mantenere elevato il valore della resistenze, al contrario della convenzionale logica NMOS o pseudo-NMOS. Dunque un processo di fabbricazione avanzato per realizzare memorie SRAM deve dare la possibilità di realizzare resistenze molto elevate per ridurre il consumo statico di potenza.

Mantenere il consumo statico il più basso possibile è una delle priorità degli attuali processi di fabbricazione delle memorie SRAM. Consideriamo una memoria SRAM da 1 Mbit alimentata a 2.5 V e con resistenza di carico dell'invertitore di 10 kΩ. Ciascuna cella assorbe una corrente di 0.25 mA in condizioni statiche e il corrispondente consumo di potenza arriva a ben 250 W. L'unica soluzione plausibile è rendere le resistenze più grandi possibili. Resistenze molto elevate, ma compatte, possono essere realizzate usando polisilicio non drogato, che ha una resistenza superficiale di parecchi TΩ/quadro (Tera = 10^{12}). L'unico scopo di questa resistenza è quello di mantenere alto il valore della tensione del nodo di uscita, compensando le correnti di perdita che tendono a scaricare l'uscita della cella (tipicamente intorno a 10^{-15} A/cella [Takada91]). Un valore di corrente così basso è raggiunto nelle attuali tecnologie impiegando transistor con tensione di soglia maggiore di quella nominale. Affinché la resistenza adempia al suo scopo, deve essere in grado di fornire una corrente di almeno due ordini di grandezza maggiore della corrente di perdita dei transistor, cioè $I_{load} > 10^{-13}$ A. Questo fornisce un limite superiore al valore della resistenza.

La considerazione che il pull-up della cella ha il solo scopo di compensare la carica persa dalla capacità del nodo di uscita, suggerisce una struttura alternativa per la cella a sei

Materiale protetto da copyright

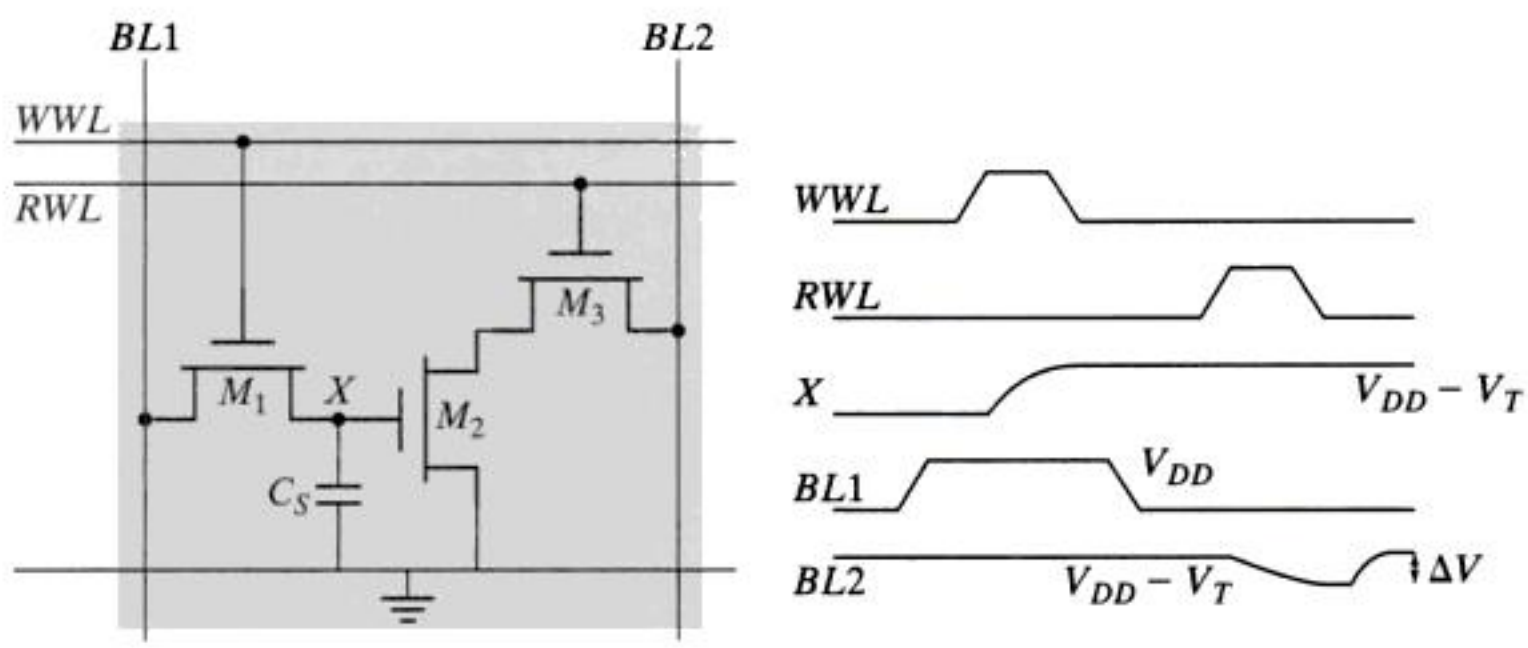


Figura 12-33 Cella di memoria dinamica a tre transistor e corrispondenti forme d'onda dei segnali durante la lettura e la scrittura.

il dato in forma sia diretta sia negata, cioè contiene ridondanza. Eliminando uno dei due transistor di pull-down (ad esempio M_1) si rimuove questa ridondanza e otteniamo la cella a tre transistor (3T-DRAM) disegnata nella Figura 12-33 [Regitz70]. Questo tipo di cella è stata la base della la prima memoria a semiconduttore a 1 Kbit realizzata da Intel [Hoff70]. Sebbene al giorno d'oggi questa struttura sia stata rimpiazzata da celle ben più efficienti in termini di occupazione di area, la 3T-DRAM rimane una cella largamente usata in molte memorie integrate all'interno dei sistemi digitali orientati ad applicazioni specifiche. Ciò è imputabile alla loro semplicità di progettazione e di funzionamento.

La cella è scritta imponendo il valore appropriato nella *BL*1 e attivando la linea *WWL* (*write-word line*). Il dato è memorizzato sotto forma di carica nella capacità C_S, detta *capacità di memorizzazione*, una volta che la tensione sulla *WWL* è stata portata a 0 V. L'operazione di lettura viene eseguita quando la *RWL* (*read-word line*) viene abilitata. Il transistor M_2 può essere acceso o spento in funzione della carica immagazzinata nella capacità di memorizzazione. La bitline *BL*2 può essere mantenuta a V_{DD} mediante un dispositivo di carico, ad esempio un transistor PMOS con il gate a massa o un transistor NMOS saturo, oppure può essere precaricata a V_{DD} o $V_{DD} - V_T$. Il primo approccio richiede di dimensionare accuratamente i transistor e produce un consumo statico di potenza. Dunque, l'approccio a precarica è generalmente preferibile. Se la capacità C_S memorizza un 1, la serie di M_2 e M_3 scarica la *BL*2 a massa, altrimenti *BL*2 rimane alta. Notiamo che la cella di memoria è invertente, cioè nella *BL*2 viene letto il valore negato del bit memorizzato. Il modo più comune per realizzare l'operazione di refresh necessita, nell'ordine, di eseguire le seguenti operazioni: la lettura della cella; l'impostazione della *BL*1 al valore complementare della *BL*2; e l'abilitazione della *WWL*.

La complessità della cella è sostanzialmente minore della realizzazione statica, come appare chiaramente guardando l'esempio di layout della cella nella Figura 12-34. L'impegno totale di area è 576 λ^2 per cella, contro 1092 λ^2 della corrispondete cella statica della Figura 12-31. In realtà l'area effettiva risulta ancora minore; infatti questi numeri non tengono conto del fatto che molti contatti sono condivisi con le celle vicine. La riduzione

Materiale protetto da copyright

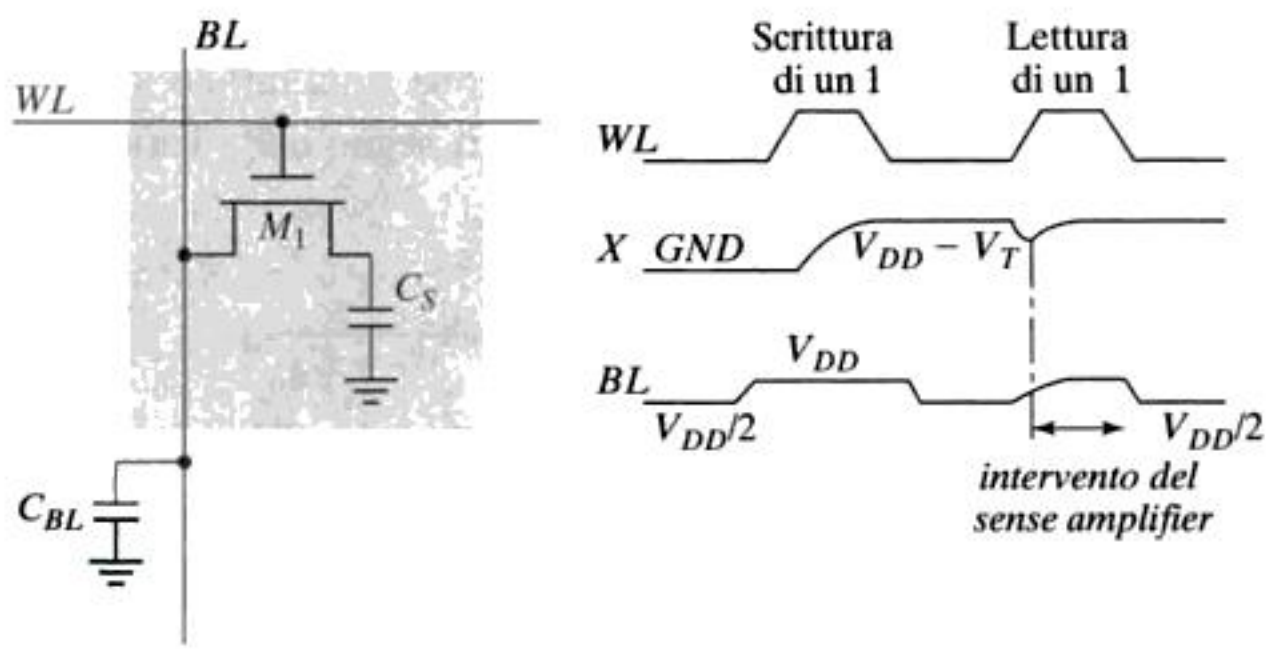


Figura 12-35 Struttura di una cella DRAM a un transistor e le corrispondenti forme d'onda durante la lettura e la scrittura.

1. Contrariamente alla cella SRAM, non esistono vincoli tra i *fattori di forma* dei transistor, che è una caratteristica comune a tutti i circuiti dinamici. La scelta delle dimensioni è dettata solo da considerazioni legate alle prestazioni e alla robustezza della cella. Vale la pena precisare che questa osservazione non è più vera se la bitline è pilotata da un semplice carico statico resistivo (sempre acceso).
2. Contrariamente ad altre celle DRAM, l'operazione di lettura della cella è *non distruttiva*, cioè il dato memorizzato non è modificato durante la lettura.
3. Non è necessario alcun passo addizionale di processo. La capacità di memorizzazione non è nient'altro che la capacità di gate del dispositivo. Ciò è una caratteristica che contradistingue questo approccio da altri tipi di celle, di cui parleremo tra breve, e che le rende molto appetibili quando la memoria deve essere integrata insieme al resto del sistema (ad esempio all'interno del microprocessore).
4. Il valore memorizzato nel nodo X quando viene scritto 1 è uguale a $V_{WWL} - V_{Tn}$. La caduta di tensione pari alla tensione di soglia, riduce la corrente attraverso M_2 durante la lettura, aumentando il corrispondente tempo di accesso. Per evitare questo problema, alcuni sistemi applicano alla wordline un impulso di tensione V_{WWL} maggiore di V_{DD}.

Cella di memoria DRAM a un transistor (1T-DRAM) Se accettiamo di sacrificare alcune delle proprietà della cella a tre transistor è possibile ridurne ulteriormente la complessità. La struttura che ne risulta è detta cella DRAM a un transistor (1T-DRAM) e rappresenta indubbiamente il tipo di cella più diffuso tra le memorie RAM dinamiche.[6] Lo schema circuitale è mostrato nella Figura 12-35 [Dennard68]. Il suo principio di funzionamento è estremamente semplice. Durante il ciclo di scrittura, il dato da memorizzare è posto nella *BL* e successivamente la *WL* viene abilitata, caricando o scaricando la capacità della cella a seconda del valore della *BL*. Prima che sia eseguita l'operazione di lettura, la bitline è precaricata alla tensione V_{PRE}. Appena la wordline è portata al valore logico alto, abilitando il pass-transistor, ha

[6]Si può immaginare anche una cella DRAM con solo due transistor. Tuttavia essa offre scarsi vantaggi sia rispetto alla cella a un transistor sia rispetto la cella a tre transistor e, quindi, viene impiegata raramente.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

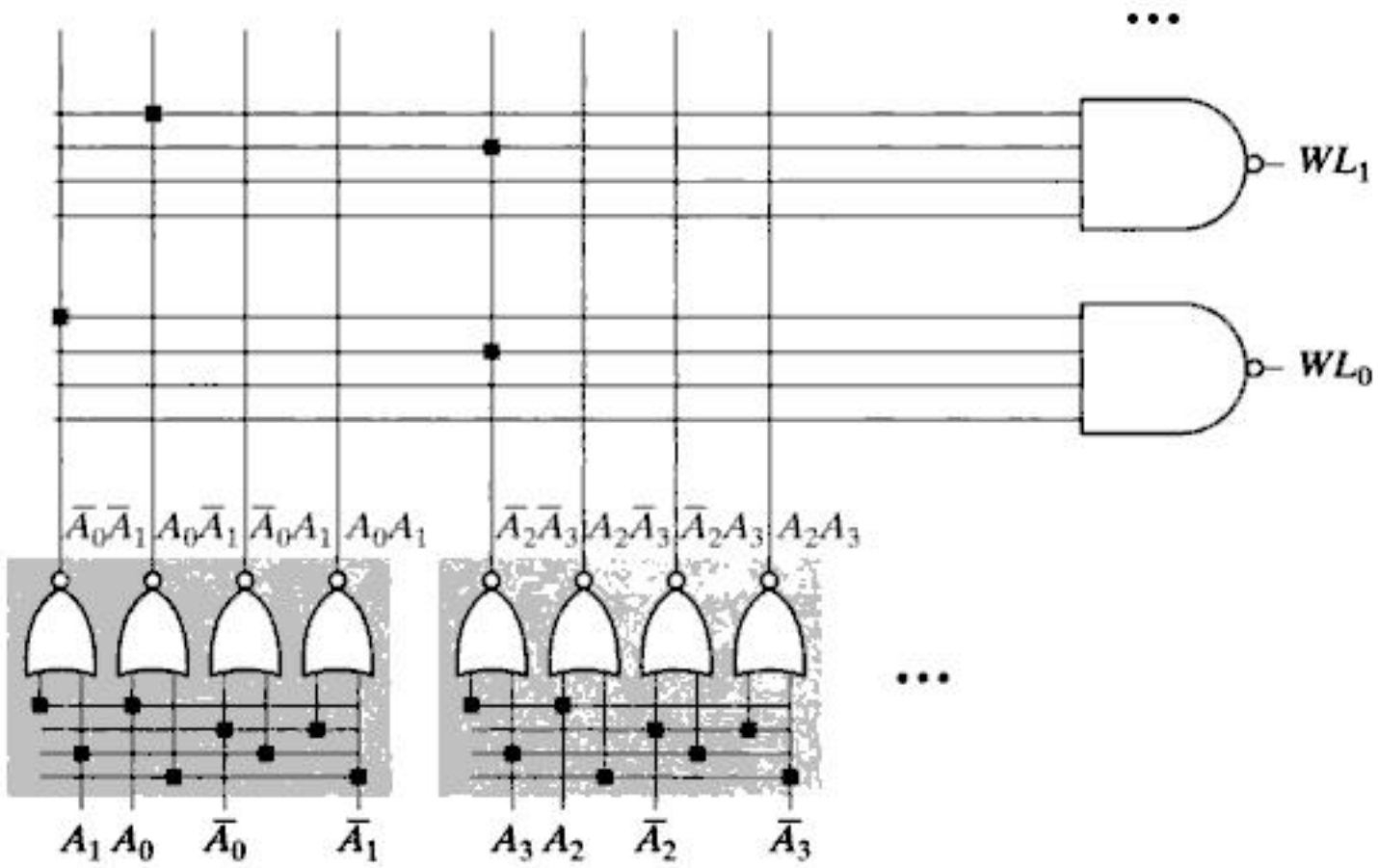


Figura 12-41 Decodificatore a NAND con predecodificatore a due ingressi.

Fortunatamente, alcuni dei principi introdotti nel Capitolo 6 ci vengono in aiuto. Dividere la porta complessa in due o più stadi logici dà luogo a circuiti più semplici, veloci ed economici. Questo concetto di scomposizione trova un largo consenso nella progettazione delle attuali memorie, rendendo possibile costruire decodificatori in tecnologia CMOS complementare molto veloci ed efficienti in termini di occupazione di area. I diversi segmenti di indirizzo sono decodificati nel primo stadio logico detto *predecodificatore*. Il secondo stadio di porte logiche combina i segmenti, generando il segnale finale di abilitazione della wordline.

Consideriamo il caso di una decodificatore a NAND con 8 ingressi. L'espressione di WL_0 può essere raggruppata nel modo seguente:

$$\begin{aligned} WL_0 &= \overline{A_0A_1A_2A_3A_4A_5A_6A_7} \\ &= \overline{(\overline{A_0+A_1})(\overline{A_2+A_3})(\overline{A_4+A_5})(\overline{A_6+A_7})} \end{aligned} \tag{12.11}$$

In questo caso particolare, l'indirizzo è suddiviso in gruppi di due bit che sono decodificati in anticipo. I segnali risultanti sono quindi combinati usando porte NAND a quattro ingressi per generare il vettore di segnali di abilitazione delle wordline della memoria. La struttura usata per il decodificatore è illustrata nella Figura 12-41.

L'impiego di un predecodificatore ha diversi vantaggi:

- Riduce il numero di transistor richiesti. Assumendo che il predecodificatore sia costruito in logica CMOS statica complementare, il numero di transistor in un decodificatore a 8 ingressi è $(256 \times 8) + (4 \times 4 \times 4) = 2\,112$, cioè il 52% di quelli necessari ad implementare il decodificatore a singolo stadio (4 096 transistor).

Materiale protetto da copyright

In questo esempio abbiamo trascurato la capacità parassita delle piste, C_W, che collegano il predecoder al decodificatore, che può essere abbastanza elevata in memorie di grandi dimensioni. Ciò è chiaro osservando l'architettura completa della memoria, come quella mostrata nella Figura 12-6. Includere questo contributo costante nell'ottimizzazione del decodificatore complica ulteriormente il problema, ma rimane tuttavia risolvibile [Amrutur01].

Infine, è possibile eseguire un'ultima ottimizzazione del circuito, osservando che normalmente le wordline hanno valore logico basso e, quindi, è possibile ottimizzare la sola commutazione basso-alto. Il lavoro logico di tutte le porte dei cammini critici possono essere sbilanciati in modo da favorire questa commutazione.

Tutti i decodificatori di grandi dimensioni sono realizzati usando un approccio a due livelli logici. Questa struttura con predecodifica ha anche un altro vantaggio: aggiungendo un segnale di selezione a ciascun predecodificatore, è possibile disabilitare il decodificatore quando il blocco di memoria non è attivo, ottenendo un considerevole risparmio energetico.

Decodificatore in logica dinamica Poiché le prestazioni del decodificatore sono determinate principalmente da una sola delle transizioni, è possibile valutare altri tipi di implementazioni. Una possibilità è la logica pseudo-NMOS che, però, è già stata scartata in precedenza a causa del consumo statico di potenza. Un'alternativa migliore viene offerta dalla logica dinamica, come nella Figura 12-43, dove è raffigurato lo schema elettrico a livello di transistor di una decodificatore 2:4. Notiamo che l'architettura di questo decodificatore ha una struttura geometricamente simile a quella di una matrice di celle ROM-NOR, differenziandosi solo per la disposizione dei transistor.

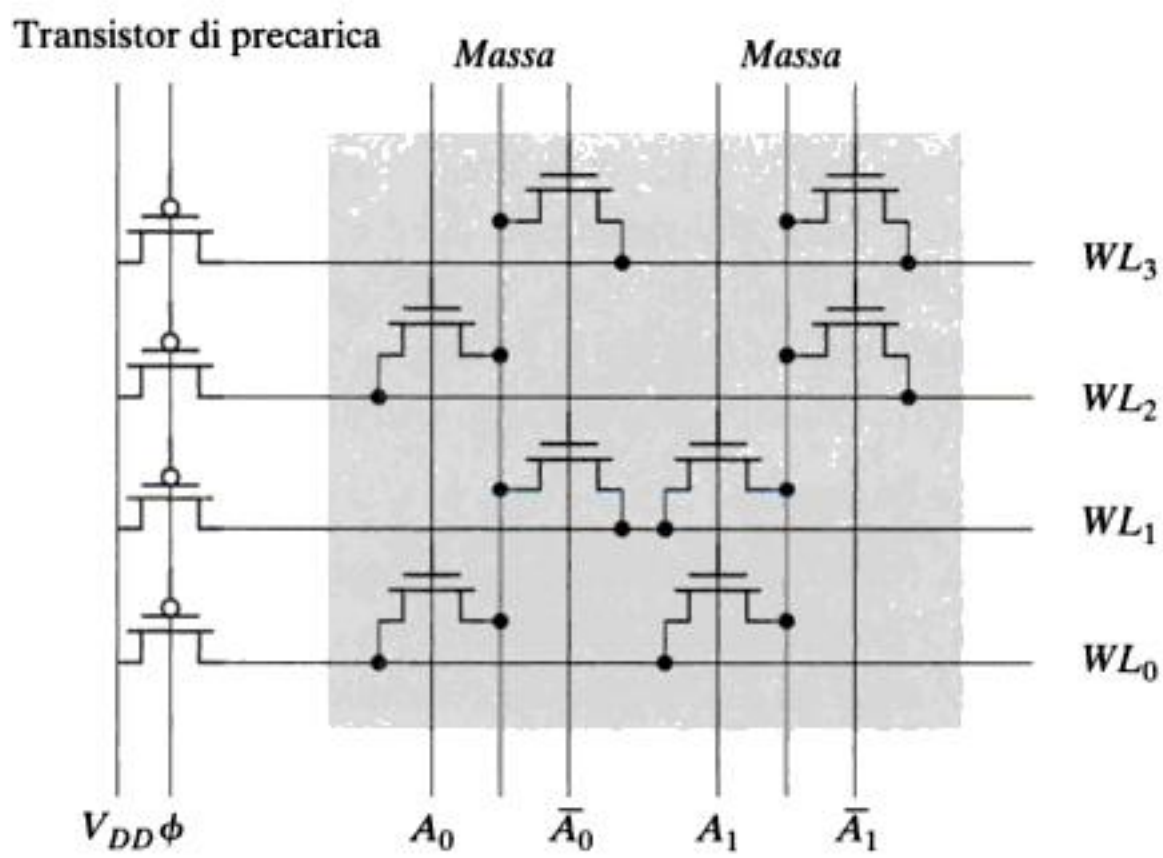


Figura 12-43 Decodificatore 2:4 dinamico a NOR.

Materiale protetto da copyright

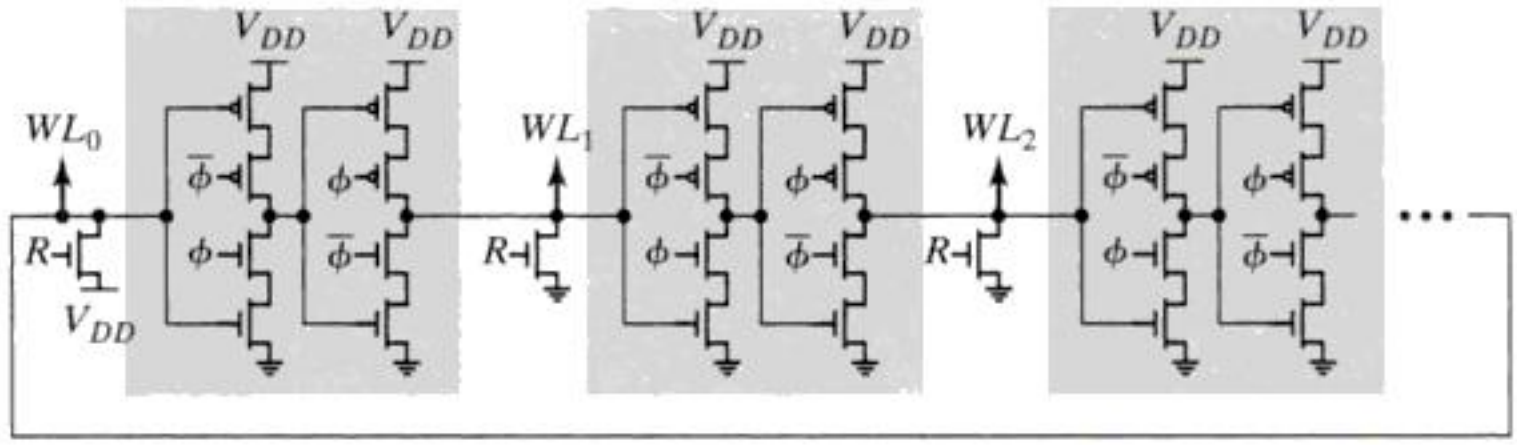


Figura 12-47 Decodificatore per memorie ad accesso seriale realizzato con un registro a scorrimento. Il segnale *R* inizializza il puntatore alla prima locazione.

12.3.2 Sense Amplifier

Il *sense amplifier* gioca un ruolo di primaria importanza sul funzionamento, sulle prestazioni, e sull'affidabilità di una memoria. In particolare, svolge le funzioni seguenti:

- *Amplificazione* — In alcune memorie, come le DRAM a un transistor, l'amplificazione è un requisito indispensabile al funzionamento stesso della memoria, poiché il valore tipico della variazione di tensione al nodo di uscita della cella è limitato a circa 100 mV [Itoh01]. In altre memorie, sebbene non sia indispensabile, il sense amplifier permette di leggere correttamente il dato memorizzato con una variazione ridotta della tensione della bitline, riducendo sia il tempo di ritardo sia il consumo di potenza.
- *Riduzione del tempo di ritardo* — il sense amplifier accelera la transizione della bitline, compensando l'elevata resistenza di uscita della cella (di dimensione minima o quasi minima) oppure rilevando una piccola variazione della tensione della bitline e amplificandola al pieno valore di escursione logica.
- *Riduzione del consumo* — Riducendo l'escursione logica del segnale della bitline è possibile risparmiare molta dell'energia dissipata per caricare e scaricare la linea.
- *Ripristino del segnale* — In memorie come la DRAM a un transistor, il sense amplifier è richiesto anche per ripristinare il valore corretto del segnale nella bitline e nella cella, durante le operazioni di lettura e di refresh.

La topologia del sense amplifier dipende molto dal tipo di memoria in cui è impiegato, dai livelli di tensione in gioco e dall'architettura complessiva della memoria. I sense amplifier sono essenzialmente circuiti di natura analogica e un'analisi dettagliata richiede esperienza nell'ambito della progettazione analogica. Dunque, nel seguito presenteremo solo una breve introduzione al progetto di un sense amplifier, mentre per maggiori dettagli rimandiamo il lettore ad altri libri di testo, quali [Gray01] e [Sedra87].

Sense amplifier differenziale

Un amplificatore differenziale riceve in ingresso una tensione differenziale di piccolo segnale (cioè, le tensioni delle due bitline) e amplifica questa variazione fornendo in uscita una singola tensione di grande segnale. È noto che generalmente l'approccio differenziale presenta numerosi vantaggi rispetto alla corrispondente configurazione ad uscita singola; uno dei più

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

accesso in lettura e scrittura è dovuto al ritardo introdotto dalle lunghe piste di wordline e bitline. Una parte considerevole dell'area occupata da una memoria è allocata per realizzare apposite strutture per pilotare le lunghe piste, in particolare i buffer di indirizzo e i buffer di ingresso/uscita. Il progetto dei buffer è già stato discusso nel Capitolo 5 e tutto ciò che è stato detto in quell'occasione mantiene la sua validità in questa particolare applicazione.

12.3.5 Temporizzazione e controllo

La discussione precedente ha presentato il modulo di memoria come un'entità molto complessa il cui funzionamento è determinato dal susseguirsi di una serie di eventi, come il campionamento dell'indirizzo, la decodifica dell'indirizzo di riga e di colonna, la precarica e l'equalizzazione delle bitline, l'abilitazione del sense amplifier e la trasmissione del segnale verso l'uscita. Affinché sia pienamente funzionante, è essenziale che tutte queste operazioni avvengano in quest'ordine sotto qualsiasi condizione e indipendentemente dai parametri tecnologici dei dispositivi. Per ottenere le massime prestazioni è necessario che tutte queste azioni siano accuratamente sincronizzate. Sebbene la circuiteria di temporizzazione e controllo occupi una porzione molto piccola dell'area dell'intera memoria, è una parte essenziale e determinante nel progetto della memoria. Essa richiede un'attenta ottimizzazione e l'esecuzione di ripetute simulazioni SPICE che tengano conto di tutte le possibili condizioni operative.

Nel tempo, sono emerse numerose tecniche di temporizzazione e sincronizzazione delle memorie che possono essere classificate in due grosse famiglie: *temporizzate* e *autotemporizzate (o asincrone)*. Di seguito, per illustrane la differenza, discuteremo un tipico esempio per ciascuna delle due famiglie.

DRAM: memorie temporizzate

Sin dalle origini, le memorie DRAM sono state caratterizzate da una procedura di accesso in cui l'indirizzo della cella viene fornito in due tempi, dove l'indirizzo di riga e quello di colonna vengono forniti in sequenza nello stesso bus di indirizzi per ridurre il numero di piedini nel package. Questo approccio è sopravvissuto a molte generazioni di memorie ed è ancora in uso oggigiorno, sebbene stia diventando sempre più scomodo e in forte contrasto con i requisiti di sistema ai più alti livelli di astrazione.

In questo tipo di accesso sequenziale l'utente deve fornire due segnali di controllo principali: *RAS (row-address strobe)* e *CAS (column-address strobe)*, che indicano la presenza rispettivamente dell'indirizzo di riga e di colonna (vedi Figura 12-8). Un ulteriore segnale (W) di controllo definisce se il tipo di accesso è una lettura o una scrittura. Questi segnali possono essere interpretati come segnali di clock esterni e hanno il compito di temporizzare gli eventi interni alla memoria. Analogamente all'approccio sincrono, i segnali *RAS* e *CAS* devono essere sufficientemente distanziati tra loro in modo che tutte le operazioni che seguono possano giungere a completamento. La Figura 12-59 mostra un diagramma temporale semplificato di una memoria DRMA a 1×4 Mbit e alcune restrizioni temporali imposte ai segnali. La descrizione completa delle specifiche temporali include più di 20 parametri, ciascuno dei quali ha limiti ben precisi per garantire il corretto funzionamento della memoria.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

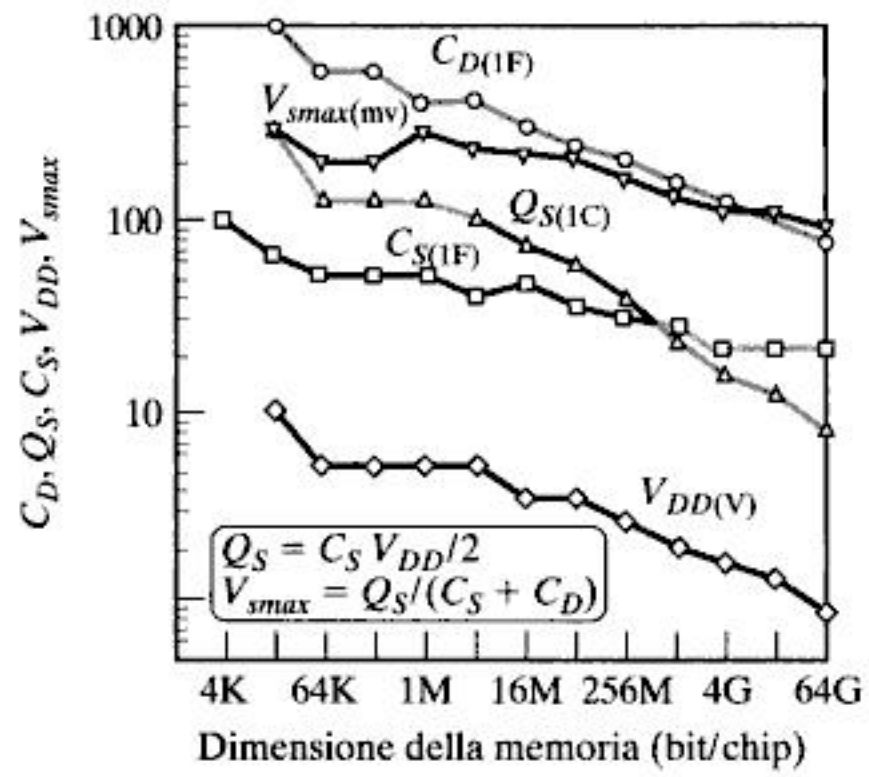


Figura 12-62 Evoluzione dei parametri delle memorie DRAM (da [Itoh01]). C_D, V_{smax}, Q_s e C_s rappresentano rispettivamente la capacità della bitline, la minima variazione di tensione rilevabile, la carica immagazzinata nella cella e la capacità della cella.

Ad esempio la capacità della cella di una DRAM è diminuita da circa 70 fF per una memoria da 16 Kb a meno di 30 fF per le attuali generazioni. Contemporaneamente, anche i livelli di tensione sono diminuiti a causa del consumo di potenza e dei problemi di affidabilità dei circuiti. Tensioni di 1 V o meno sono la norma per le memorie di nuova generazione. La ragione principale di ciò è la necessità di limitare i processi di degradazione che devono essere tollerati dal sottile ossido di gate dei transistor della cella. In conseguenza di tutto ciò, la carica immagazzinata in una cella di DRAM è diminuita di un fattore 5 passando dalla generazione dei 64 Kb a quella dei 64 Mb. Il fatto che la carica sia diminuita di un fattore 5 contro il fattore 100 di cui è stata ridotta l'area della cella in questo periodo, è dovuto soltanto alle innovazioni introdotte sull'architettura della cella stessa.

Allo stesso tempo, l'accresciuta densità di integrazione, fa aumentare il livello del rumore a causa della maggiore interferenza tra i segnali. Sebbene il problema dell'accoppiamento capacitivo tra wordline e bitline fosse già un problema noto nei primi anni '80, la realizzazione di piste sempre più vicine ha portato alla luce altri problemi come l'accoppiamento tra le bitline vicine. A contribuire al problema si sommano anche l'alta velocità che accresce il rumore derivante dalle commutazioni in ogni nuova generazione di memorie, e gli errori software indotti da particelle α (*soft error*), che possono alterare lo stato di un nodo ad alta impedenza in modo completamente casuale, comportandosi come una sorgente addizionale di rumore. Quest'ultimo problema, originariamente peculiare delle memorie dinamiche, si sta ora propagando anche alle RAM statiche, poiché l'impedenza dei nodi di segnale sta aumentando sempre di più anche in queste memorie. La Figura 12-63 elenca le sorgenti di rumore che influenzano il comportamento di una cella DRAM a un transistor, la maggior parte delle quali sarà discussa tra breve nei successivi paragrafi.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

mentando notevolmente il consumo statico di potenza della memoria. L'introduzione di nuove tecniche sta diventando una necessità reale e concreta.

12.5.1 Origine del consumo di potenza nelle memorie

Il consumo di potenza nelle memorie può essere attribuito a tre contributi fondamentali: la matrice di celle, i decodificatori (di riga, di colonna e di blocco) e la circuiteria periferica. Un'equazione unificata valida per una memoria CMOS con m colonne e n righe è stata fornita in modo approssimato da [Itoh01] (vedi Figura 12-70). per un normale ciclo di lettura:

$$\begin{aligned} P &= V_{DD} I_{DD} \\ I_{DD} &= I_{matrice} + I_{decodificatori} + I_{periferia} \\ &= [m i_{act} + m(n-1) i_{hld}] + [(n+m) C_{DE} V_{int} f] + [C_{PT} V_{int} f + I_{DCP}] \end{aligned} \tag{12.19}$$

Nell'equazione sono stati usati i seguenti parametri:

- i_{act}: la corrente assorbita dalle celle selezionate (o attive).
- i_{hld}: le corrente assorbita dalle celle non attive necessaria per la ritenzione del dato.
- C_{DE}: la capacità del nodo di uscita di ciascun decodificatore.
- C_{PT}: la capacità totale dei circuiti logici periferici CMOS.
- V_{int}: la tensione di alimentazione interna.
- I_{DCP}: la corrente statica (o quasi statica) della circuiteria periferica. La maggior parte di questa corrente è dovuta ai sense amplifier e alla circuiteria presente in ciascuna colonna. Altre sorgenti provengono dai generatori di tensioni interni al chip.
- f: la frequenza operativa.

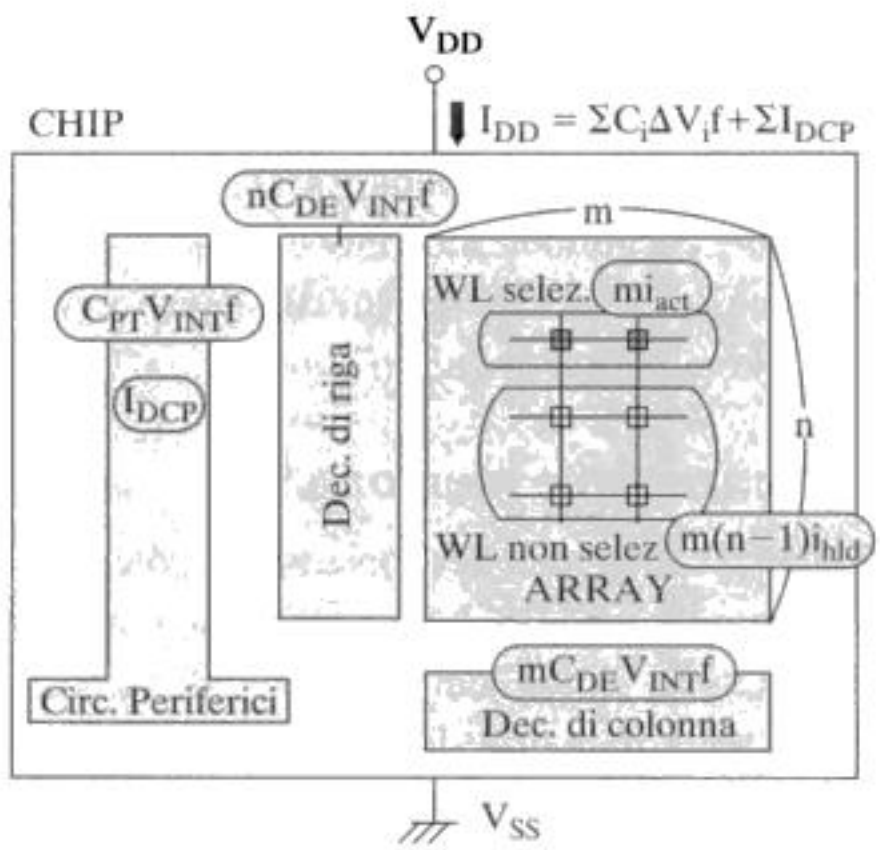


Figura 12-70 Origine del consumo di potenza in una memoria CMOS [Itoh01].

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

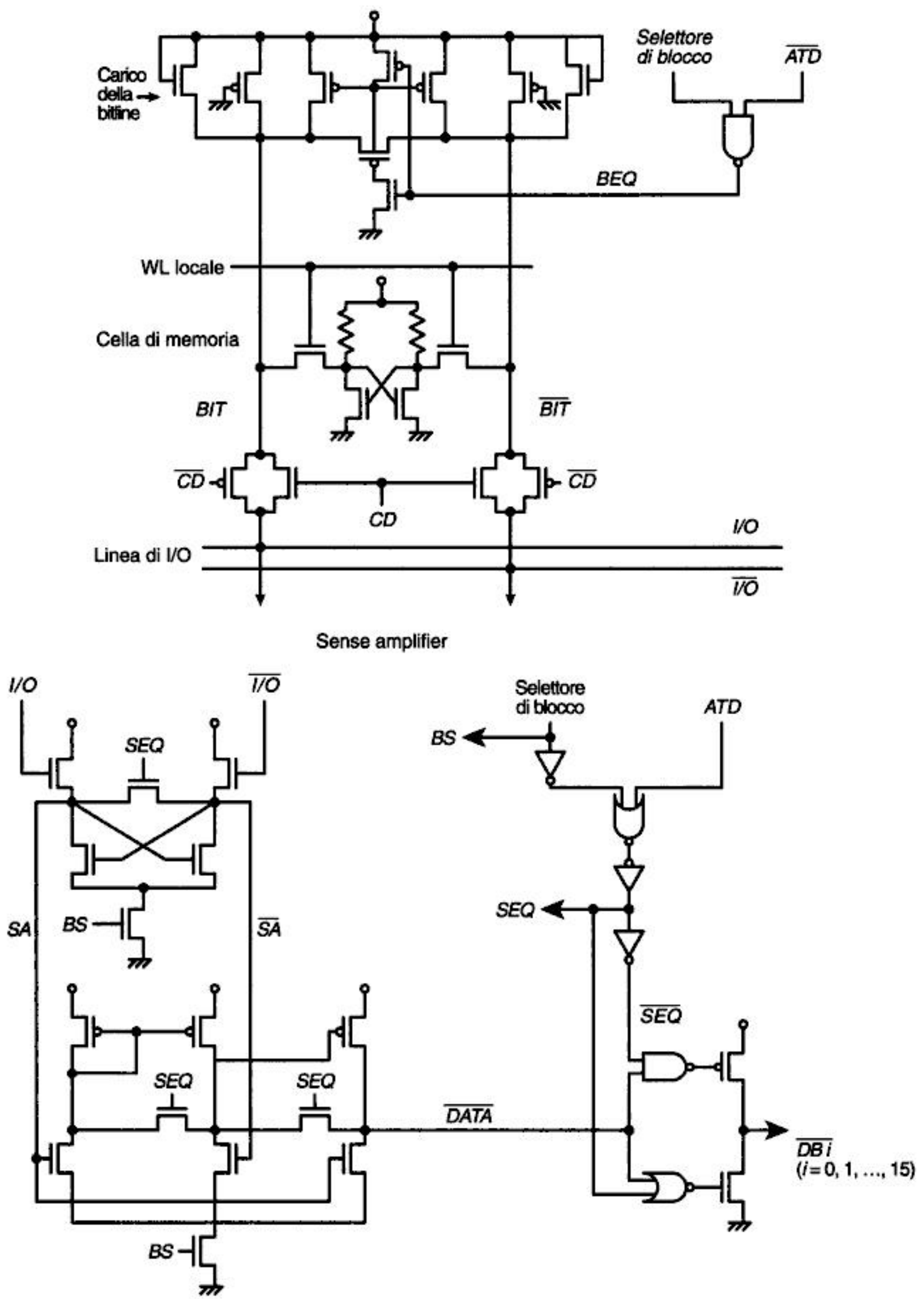


Figura 12-78 Circuiti periferici della bitline e corrispondenti forme d'onda dei segnali.

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

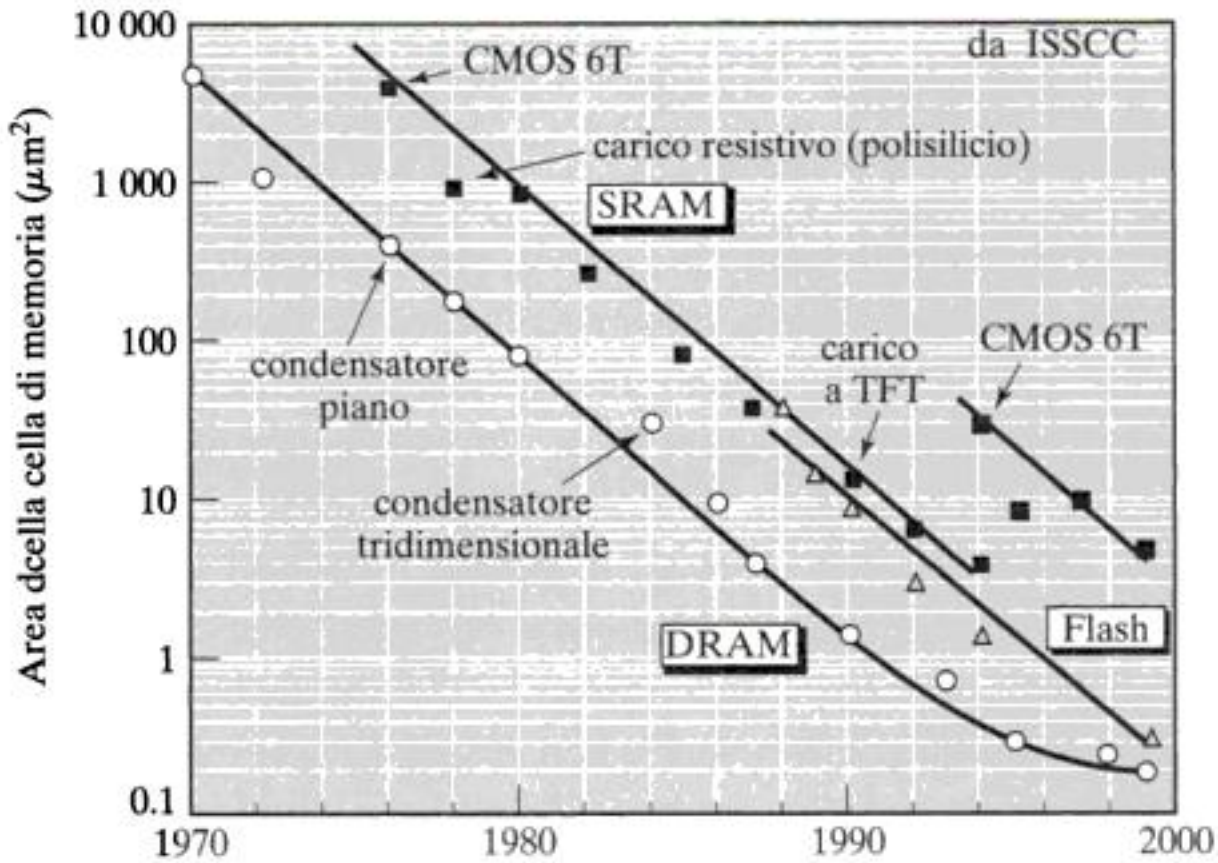


Figura 12-83 Evoluzione dell'occupazione di area delle celle di memoria VLSI [Itoh01].

È interessante capire i fattori che hanno promosso un'evoluzione così spettacolare. La Figura 12-83 raffigura l'evoluzione nel tempo delle dimensioni della cella per le diverse generazioni di memorie. Le dimensioni delle celle DRAM e SRAM sono state ridotte ad un passo di un quindicesimo ogni 10 anni (ovvero un fattore tre o più ogni tre anni). In aggiunta, le dimensioni del chip sono cresciute di un fattore 1.4 ad ogni generazione. Dalla combinazione di questi due fattori otteniamo un aumento di un fattore quattro del numero di celle per ciascuna successiva generazione. Verrebbe spontaneo attribuire la riduzione delle dimensioni della cella solamente al fattore tecnologico, ma ciò non sarebbe completamente corretto. Ad esempio, la lunghezza effettiva del canale del transistor, L_{eff}, usati in una memoria SRAM a 4 K è di 3 μm ed è diminuita di generazione in generazione fino ai 0.25 μm usati nelle memorie SRAM da 64 M. Questo fa pensare che la miniaturizzazione dei dispositivi abbia contribuito alla riduzione delle dimensioni della cella per un fattore di $1.5 \times 1.5 = 2.25$ tra una generazione e la successiva, che non è sufficiente a spiegare la tendenza reale. Un'ulteriore riduzione delle dimensioni è stata ottenuta grazie alla maggiore sofisticazione della tecnologia di fabbricazione delle memorie: tra i più importanti miglioramenti ricordiamo l'introduzione del gate in polisilicio, l'isolamento a LOCOS e STI, i carichi resistivi in polisilicio per le celle SRAM e la struttura tridimensionale delle celle DRAM che impiegano condensatori verticali a trincea o a pila. L'introduzione continua di queste innovazioni è essenziale affinché questo passo di evoluzione possa essere mantenuto anche nei prossimi anni.

Mantenere il passo potrebbe non essere un problema così banale, poiché negli ultimi anni sono emersi numerosi problemi e impedimenti. Questo è già evidente nella Figura 12-82, che mostra l'inizio della saturazione nella capacità delle memorie SRAM e DRAM alla fine degli anni '90. Di fatto, le memorie Flash stanno per raggiungere le DRAM, come abbiamo già avuto modo di vedere in precedenza. La saturazione delle dimensioni delle celle è principalmente dovuta ai costi elevati e ai problemi tecnologici in cui si incorre nel miniatu-

Materiale protetto da copyright

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

You have either reached a page that is unavailable for viewing or reached your viewing limit for this book.

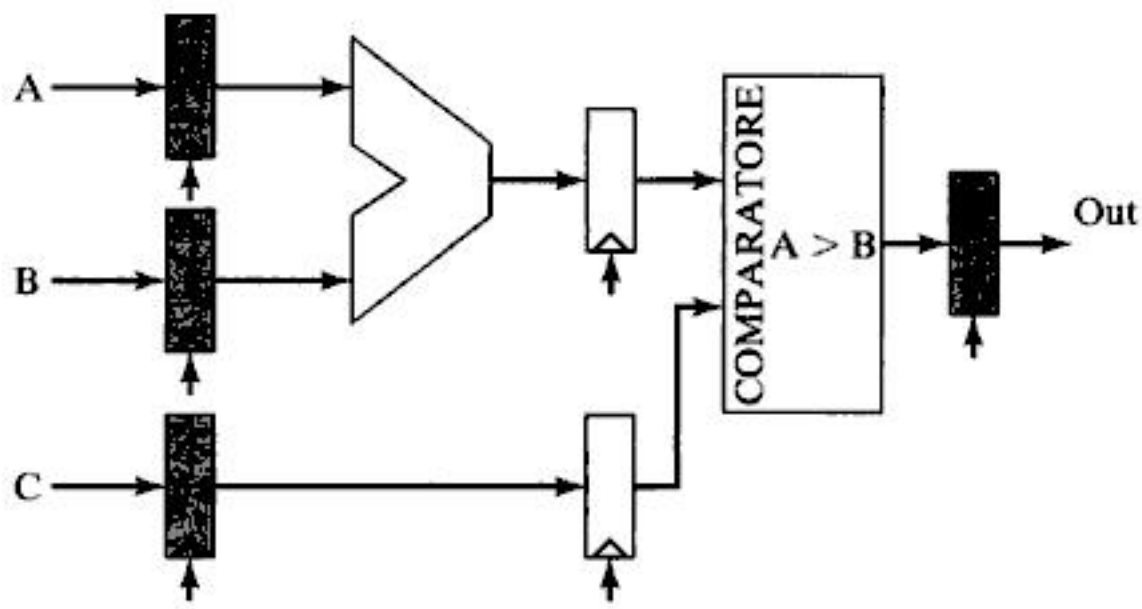


Figura H-8 Cammino dati con struttura pipeline che impiega la scansione parziale. Solo i registri di colorati in grigio sono inclusi nella catena di scansione.

test, il sommatore e il comparatore possono essere considerati come un singolo blocco combinatorio. La sola differenza è che in fase di esecuzione sono necessari due cicli di clock per trasmettere i dati di ingresso attraverso la rete combinatoria ai registri di uscita. Questa tecnica è detta *scansione parziale* ed è spesso impiegata quando le prestazioni sono il parametro di massimo interesse. Lo svantaggio è che la decisione su quali registri inserire nella catena di scansione non è sempre ovvia e richiede l'intervento del progettista.

H.3.4 Tecnica di scansione al contorno

Fino a poco tempo fa, i maggiori problemi di collaudo provenivano dal test del circuito integrato. Il collaudo delle schede era di gran lunga facilitato dall'abbondante disponibilità di terminali di test sulla scheda. Montando il chip sulla scheda attraverso i fori, ciascun piedino è accessibile (controllabile e osservabile) dal lato posteriore della scheda stessa. Per il collaudo bastava impiegare un sistema di sonde sul quale veniva appoggiata la scheda in modo tale da contattare tutti i nodi dei segnali di interesse. La situazione è cambiata radicalmente dal momento in cui sono state introdotte le tecniche di *packaging* avanzate, come il montaggio superficiale o i moduli multichip (vedi Capitolo 2). Controllabilità e osservabilità non sono così immediatamente raggiungibili, poiché il numero di punti di accesso ai segnali nella scheda è diminuito radicalmente. Per affrontare il problema è possibile estendere il metodo di test a scansione dal componente alla scheda.

Il tipo di approccio che ne consegue è detto *scansione al contorno (boundary scan)* ed è stato standardizzato per assicurarne la compatibilità tra i diversi costruttori ([IEEE1149]). La procedura consiste nel connettere i terminali di ingresso e uscita dei diversi componenti montati sulla scheda in una catena di scansione seriale, come rappresentato nella Figura H-9. Durante il normale funzionamento, i terminali della catena di scansione fungono da ingressi e uscite dei dispositivi. In modalità di test, è possibile leggere e scrivere i vettori di test attraverso la catena, rendendo controllabile e osservabile il componente al contorno (da qui il nome della tecnica). L'operazione di test procede in maniera analoga a quanto descritto nel paragrafo precedente. Varie strutture di controllo consentono il test del singolo componente o delle interconnessioni sulla scheda. Tra i costi supplementari di questo approccio vi sono

Materiale protetto da copyright

Jan M. Rabaey
Anantha Chandrakasan
Bora Nikolić

Circuiti integrati digitali

L'ottica del progettista

Seconda edizione

Edizione italiana a cura di Andrea Cester e Andrea Gerosa - Università di Padova

Indicato per i corsi di elettronica dei sistemi digitali sia nelle lauree triennali sia nelle lauree specialistiche in Ingegneria elettronica e Ingegneria dell'informazione.

In questa seconda edizione di uno dei testi più apprezzati sulla progettazione di sistemi digitali, gli autori dedicano ampio spazio alla trattazione estesa di tematiche quali i dispositivi submicrometrici, l'ottimizzazione circuitale, la modellizzazione ed ottimizzazione delle interconnessioni, l'integrità del segnale, la temporizzazione, la sincronizzazione e il consumo di potenza. Tutti questi argomenti vengono discussi insieme ad esempi progettuali in linea con lo stato dell'arte. Data l'importanza delle metodologie negli odierni processi di fabbricazione, sono stati inclusi nel testo degli inserti sulle metodologie progettuali, ciascuno dei quali pone in risalto un particolare aspetto del flusso di progettazione.

Ogni capitolo segue un approccio coerente: vengono presentati prima i concetti chiave a livello teorico, quindi ne viene mostrata l'implementazione pratica attraverso l'utilizzo di casi reali. Sempre a supporto della didattica, alla fine di ogni capitolo, viene posta una serie di esercizi e problemi (per un totale di circa 500) che consentono non soltanto di verificare il livello di comprensione dello studente, ma anche di proporre un momento di confronto attraverso l'analisi di casi reali.

Supplementi
A corredo del testo è stata resa disponibile una vasta gamma di materiali di supporto liberamente scaricabili dal sito Web dell'autore (http://bwrc.eecs.berkeley.edu/IcBook/).

Per i docenti e gli studenti:
- Raccolta di esercizi
- Problemi di progettazione
- Layout ed esempi hardware e software per il laboratorio
- Software di progettazione

Per i docenti:
Le trasparenze in italiano per la preparazione delle lezioni sono disponibili (facendone richiesta all'editore - hpeitalia@pearson.com) solamente per i docenti che adottano il libro come testo ufficiale del corso.

Gli autori
Jan M. Rabaey - Department of Electrical Engineering and Computer Sciences - University of California, Berkeley.
Anantha Chandrakasan - EECS Department - Massachusetts Institute of Technology.
Bora Nikolić - Department of Electrical Engineering and Computer Sciences - University of California, Berkeley.

ISBN 88-7192-231-X
9 788871 922317

PEARSON
Education Italia

Visitate il nostro sito web
http://hpe.pearsoned.it

€ 53,00

Materiale protetto da copyright